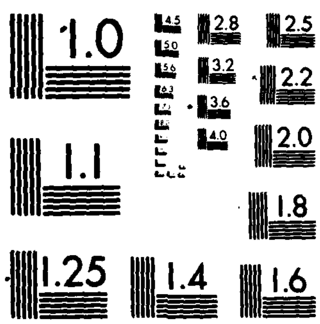
AD-A083 713 AIR FORCE INST OF TECH WRIGHT-PATTERSON AFB OH SCHOO--ETC F/G 14/1
AN ANALYSIS OF THE RCA PRICE-S COST ESTIMATION MODEL AS IT RELA--ETC(U)
DEC 79 R E STEFFEY
UNCLASSIFIED AFIT/GSM/SM/79D-20 NL

1 of 3
AD
A083713



MICROCOPY RESOLUTION TEST CHART

AN ANALYSIS OF THE RCA PRICE-S COST ESTIMATION MODEL AS IT RELATES TO CURRENT AIR FORCE COMPUTER SOFTWARE ACQUISITION AND MANAGEMENT

Thesis

Raymond E. Steffey Jr.
GSM/SM/79D-20 Captain USAF

AFIT/GSM/SM/79D-20

AN ANALYSIS OF THE RCA PRICE-S COST ESTIMATION MODEL AS IT RELATES TO CURRENT AIR FORCE COMPUTER SOFTWARE ACQUISITION AND MANAGEMENT.

THESIS

Presented to the Faculty of the School of Engineering
of the Air Force Institute of Technology
Air University
In Partial Fulfillment of the
Requirements for the Degree of
Master of Science

by

Raymond E. Steffey Jr
Captain USAF

Graduate Systems Management

Dec[redacted]79

Approved for public release; distribution unlimited.

Preface

One of the most highly visible areas of management concern in the Air Force today is computer software acquisition costs. Software acquisition throughout the Air Force have been marked by cost overruns, late deliveries, poor reliability, and user dissatisfaction in meeting stated requirements. A great deal of study and analysis of computer resource acquisition and management has been accomplished in the recent past. Advances in project planning and organization techniques, design methodologies, coding and testing practices, use of programming support tools, documentation standards, and configuration management procedures and practices hold promise for ultimately reducing Air Force expenditures on computer software.

One area which has not received quite as much emphasis as software acquisition and management is the area of software cost estimation. The Air Force requires quantitative information about the effects of current programming practices on software development costs. At the present time, there are a number of estimating techniques that are alleged to result in increased software cost estimation reliability. A continuing repository and data collection system combined with a standardized cost estimation methodology and procedure would go a long way toward resolving the difficulties encountered

in obtaining accurate software development cost and schedule estimates. The Air Force could thus encourage contractors and in-house development agencies to employ standardized recording and estimating practices.

This research was initiated in an effort to enable software cost analysts, as well as managers, to more accurately predict the cost of software development projects. I would like to take this opportunity to thank the many people who contributed to this effort. Messrs. F. Frieman, R. Park, and C. Mauro from RCA Price Systems deserve special thanks for their help in understanding and using the PRICE-S model. Thanks are also due to Captain R. Hickcox - ESD, Captain T. Landry - AU, Lt. R. Christie - AFDSDC and Mr. C. Houlette - AFLC for their assistance in gathering the data utilized in this research. Many thanks must go to Mr. D. Ferens - ASD, who provided data and arranged for the computational resources necessary to complete this research. In addition, I would like to thank Dr. C. McNichols, my advisor, and Dr. J. Cain for their invaluable assistance in preparing this report.

To my wife, Sharon, and daughters, Laura and Nicole, thanks are not enough.

Contents

List of Figures

Figure		Page

<u>List of Tables</u>

AFIT/GSM/SM/79D-20

Abstract

The enormous technical accomplishments of the computer industry have led to the building of computers of all sizes and complexities. As the range of defense computer applications grows and the complexity of the tasks these systems are called upon to handle increases, the costs of developing the application software has also increased such that it has now become a dominant component in the total system cost.

Many software acquisitions have experienced cost and schedule overruns leading to unanticipated cost growth. These experiences have highlighted the need to improve methods of software cost estimation. Software cost estimation is essential to budgeting, allocation of resources, and control of expenditures throughout the life cycle of a system. Accurate predictions of software costs are required in order to make practical and realistic tradeoffs between system capabilities and life cycle costs.

The purpose of this research is to provide those involved in the software cost estimation task with an introduction to Air Force computer resource acquisition and management in general; and specifically to investigate the applicability of the RCA PRICE-S software cost estimation model to Air Force applications system development. A mass of computer software acquisition and management study, policy, and guidance

literature was reviewed, and an attempt was made to consolidate the most pertinent information into a description of the overall processes. Historical cost and schedule data were collected on 18 Air Force software development projects. This data which included systems of the three major Air Force applications areas of: 1) embedded avionics; 2) embedded command and control; and 3) management data systems was used to calibrate and validate the PRICE-S cost estimation model.

Based on the data available in this preliminary analysis effort, it appears that the PRICE-S model is compatible with current Air Force software acquisition and management techniques. A system such as the PRICE-S system, combined with an adequate data collection methodology, might be successfully implemented giving the Air Force the capability to accurately predict and track future software development costs across the entire spectrum of software applications.

AN ANALYSIS OF THE RCA PRICE-S COST ESTIMATION MODEL AS IT RELATES TO CURRENT AIR FORCE COMPUTER SOFTWARE ACQUISITION AND MANAGEMENT

I. Introduction

The use of computer technology in the Air Force has evolved along three parallel paths: scientific, management, and operational. In any attempt to define the problems facing the Air Force in the computer area, the pervasiveness of the technology becomes immediately and readily apparent. Practically every area of Air Force operations today is involved with some facet of computer technology. The explosive growth in the use of computer technology over the past twenty years has been segmented and largely unplanned in an overall sense.

Since the Air Force is organized along functional lines, it might be expected that computer systems would also grow along functional (vertical) lines. They did. Today, approximately 33 percent of our inventory is devoted to Management Information Systems; 21 percent to Command, Control, and Intelligence; 23 percent to Scientific and Engineering; 13 percent to Logistics and Maintenance; and 10 percent to Avionics applications (Ref 2: 30). Each functional area developed its own expertise to exploit the technology to best serve its functional needs. Today's computers, when effectively coupled with communications,

enable systems to be organized vertically (WWMCCS); horizontally (Base Level B-3500 multipurpose system); or integral to a subsystem which is, in turn, a part of a larger system (F-15 Automatic Test Equipment). Some systems may even be designed in horizontal and vertical combinations.

Today, focal point responsibility for computer system development efforts reside with the Comptroller Function for Management Information Systems, a majority of the Logistics and Maintenance Systems, and some Command and Control Systems. The Intelligence community is responsible for some of the Command and Control and Intelligence Systems, while the Research and Development Function is responsible for the Scientific and Engineering, Avionics, and a number of the Command and Control Systems. The lack of a single planning approach has been unavoidable because of the differences in acquisition methodology for general purpose versus special purpose computer resources.

Computer Utilization

Within the Air Force computer utilization can be divided into two separate administrative and functional parts, operationally deployable systems and information management systems. Common to both areas of application, however, are the computer installations and physical facilities necessary to create, develop, test and evaluate, and subsequently support the computer software regardless of the nature of the application.

Operationally deployable or embedded computer systems, as they have come to be known, are generally special purpose

machines used in such functions as:

1. radar and navigation
2. weapons delivery
3. training and simulation
4. command, control and communications (battlefield)
5. scientific data gathering
6. automatic testing

Information management systems are usually general purpose, commercially available computers used for the functions of:

1. resource and inventory management
2. command, control and communications (executive level)
3. intelligence data gathering
4. general information services
5. modeling and simulation

The earliest applications of computers in the Air Force were for scientific purposes, primarily to exploit the mathematical capability of the computer. The SAGE system was the first large-scale operational use of computers integrated with weapons, sensors, facilities, and communications (Ref 4:3). The success of the SAGE program led to a greater demand for more sophisticated computer-based systems throughout the Air Force. Management interest at that time centered on the computers and their applications available and not on potential future operational requirements. Initially, Congress, DOD, and Air Force policy and organization was developed around the "commercial" information management systems.

Broad usage of digital computers in operational systems (embedded, special purpose) did not come about until technologies were developed to overcome the prohibitive weights, volume, cooling, reliability, and maintainability problems associated with the earlier computers. The time lag for state-of-the-art

development of transistor and microelectronic circuitry which allowed computers to be employed in operational systems resulted in a corresponding delay in DOD and Air Force Management attention to the development of policy, planning, and support of the embedded systems in a manner similar to that developed in the Comptroller Function for the information systems.

Until the past three to five years, ADP managers were almost completely hardware oriented. Today, however, more emphasis is being placed on the software aspects, and the computer philosophy is becoming more system oriented. Officials are slowly finding out that hardware is not the dominant force in successful system development. The "long pole in the tent" has become software. The average price of a computer in 1974 had been reduced from $3 million to $375,000, and the cost of 100,000 calculations from $25 to .009 mills (Ref 11:6). But the cost to write one instruction has steadily increased to approximately $75 per instruction to develop and $4,000 per instruction to maintain for avionics software (Ref 19:4). Software complexity and costs have continued to increase while the cost of hardware has continued to decline. In 1953, hardware accounted for 80 percent of the total cost of a system; today, it is about 25 percent and is projected to be less than 10 percent by 1985 (Ref 32:3). Operational costs, which include the maintenance of systems after development, have increased from 12 to 40 percent.

The estimated annual ADP costs in DOD are $2.9 billion to $3.6 billion for software and a total $6.2 billion to

$8.3 billion when hardware and other ADP resources are included. This is approximately 30 to 50 percent of all electronics costs in DOD (Ref 18:16). The Air Force's share of the DOD ADP budget is estimated at 35 percent of the total. In 1976, there were approximately 115 major defense systems exhibiting critical computer dependency (Ref 19:4).

Related Studies

The recognition of ADP management problems in the Air Force is not a recent happening. This section will present a brief summation of the major findings and recommendations from three of the many recent studies in the area of Air Force management.

The Air Force Command and Control Information Processing in the 1980s (CCIP-85) study in 1974 (Ref 32:4) suggested that relative to overall Air Force Command and Control operations, automated information processing capabilities will assume much more significance by the 1980s. The computer resources used will be required to operate in a highly changeable, unpredictable, and hostile environment; and critical outages or mistakes would affect national survival. The major findings of the study concluded that software is unreliable, is the major cause of program slippages, is frequently unresponsive to requirements, and will be the major strain on the Air Force ADP budget. Recommendations contained in the report included:

1. provision of R&D guidelines for development of more versatile, more economical, and less manpower intensive C&C systems;

2. reduction of the typical C&C information processing system development time from 4 to 6 years, and the resulting hardware age at IOC from 3 or 4 years to 1 or 2 years;

3. development of a software-first machine strategy, or provision of a computer with microprogramming capability which would allow it to simulate a range of hardware configurations, thus allowing the Air Force to develop C&C software before having to make an irrevocable commitment to a particular hardware configuration.

The study suggested that serious management problems and institutional roadblocks must be addressed. Procurement and configuration management practices would need major reorientations to reflect the increasingly dominant role of software, technical advances in hardware architecture, and innovations such as structured programming and software-first machine.

The study of military electronics, known as Electronics X (Ref 23:14), concluded that the major causes of excessive costs and delays were the selection of a too small or improper central processor for the system, program overintegration, lack of discipline in system development. Recommendations from the study included:

1. completing the design of the system and the basic program structure in substantial detail before making major commitments to hardward and coding

2. selecting a processor of adequate size, writing

highly modular programs emphasizing structure and overall efficiency.

3. using standard, well established programming languages

Recommendations from the study included: 1) the use of system-function-oriented hardware structures as opposed to centralized programmable uniprocessors; 2) selection and development of a processor design that will minimize the combined costs of hardware and software; 3) detailing of system design and evaluation of alternate processor architectures before hardware selection; and 4) standardization of formats and speeds for data interchange among sensors, processors, controls, and displays.

The Project Pacer Flash Study (Ref 8:48) was established to assess alternative methods of providing support for weapon system computer resources. The major conclusions and recommendations of the study were:

1. an increase in Air Force organic support of weapon system software could increase responsiveness and decrease cost
2. an organic capability for dynamic simulation and verification/validation of airborne weapon system software is required
3. adequate documentation for weapon system computer resource support is not being provided by the contractors or acquired during the acquisition phase
4. software must be accorded the same degree of management control accorded hardware, and management systems and configuration management procedures must be

evolved to support this concept

5. Air Force directives require revision, expansion, or new issue to adequately cover the weapon system computer resource acquisition and support problems.

Finally, the Tactical Computer Software Acquisition and Maintenance Study (Ref 13:50)found that :

1. Congress advocates efficient management of ADP resources while OMB and OSD management policies do not cover those resources.
2. Failure to recognize the maintenance function early enough yielded late and inadequate contractor documentation, ineffective configuration management, lack of standards, and multi-million dollar integration facilities.
3. The Comptroller manages ADP resources while DDR&E, ASD(I&L), and ASD(T) are concerned with acquisition, use, and maintenance of tactical digital computers and software.
4. Four different documents defined software documentation standards for the services, and configuration management directives were hardware oriented.

Some of the rather far-reaching recommendations of this study included the:

1. education of top management as to the effect of digital computers and software on tactical system acquisitions and life cycle support;
2. review of DOD organizational responsibilities for

computer resource acquisition, use, and maintenance;

3. issue of policies covering the use of standard computers and software languages.

These are but a few observations of the many, perhaps hundreds, of studies over the recent past which have concentrated on the computer resource area. It can be seen that time and again the same problems have been recognized. Excessive development and maintenance costs, scheduled slippages and delays, excessive errors or faults, and duplication and lack of standardization are among the most prevalent.

Embedded Computer Resource Acquisition

An embedded computer is defined as a computer which is integral to a combat weapons system when physically incorporated into the weapon system, or integral to the weapon system from a design, procurement or operations viewpoint. Being integral to means being dedicated to and essential in real time to the performance of the mission of the weapons system in combat. A combat weapons system is an instrument of combat, either offensive or defensive, used to destroy, injure, or threaten the enemy (Ref 17:3). The purpose of developing this definition was to maintain in the system program offices the full responsibility for the combat weapons systems in which computers are subordinate elements, thus excluding them from the Congressional, DOD, and Air Force directives governing general purpose ADP equipment.

A contract is the basic method used by DOD to procure equipment, supplies and services. Contract award and performance

are controlled by the Armed Service Procurement Regulations (ASPRs). The basic authority to procure equipment, supplies, and services is contained principally in Title 10, United States Code. DOD Directive 5000.1, "Major Systems Acquisition," is the primary weapon system acquisition policy directive. It should be pointed out that although computer resources may be absolutely critical to the overall operation of a weapon system, it is not usually the primary part of the system. The computer is only one of many dissimilar configuration items that make up the major system. For management purposes, weapon system acquisition programs are categorized as follows:

1. Major Program - $75 million RDT&E, $300 million production;
2. AF Designated Acquisition Program - $50 million RDT&E, $200 million production;
3. Small Program - less than $50 million RDT&E, and $200 million production (Ref 13:6).

The majority of all weapon system acquisitions fall into the small category. The primary difference in the acquisition programs is in the level of review.

AFR 800-2, "Program Management," is the basic Air Force document which interprets and implements DOD 5000.1 policies (Ref 1:12). Initially, a Statement of Operational Need (SON) is prepared which provides the basic justification to initiate new systems acquisitions. AFR 57-1, "General Operational Requirements," defines the requirements process. SONs are submitted to HQUSAF/RD and validated by the HQUSAF Requirements

Review Group (RRG). RRG validation for a small program constitutes HQUSAF approval of the SON, permits the Air Force to commit resources, and directs the implementing command (usually AFSC) to explore alternatives. AF designated or Major programs require additional approvals at the Secretary Air Force and Secretary Defense levels respectively (Ref 1:4).

Upon approval of the requirement, action is initiated to enter the system into the DOD Planning Programming and Budgeting System (PPBS). The PPBS is the means by which service needs are communicated to Congress. A key output of the PPBS is the Five-Year Defense Program (FYDP) which lists the current approved program. The Congressional Appropriations Bill makes funds available to proceed with a program. The majority of weapon systems acquisitions, of which the computer resources are an integral part, are funded under the 3600-RDT&E, 3010-Aircraft Procurement, and the 3080-Other Procurement appropriations (Ref 30:48).

The official USAF document used to provide direction is the Program Management Directive (PMD). The PMD is issued to a field product organization (usually ASD, ESD, SAMSO) which assigns acquisition responsibility to a new or existing System Program Office (SPO). The Air Force normally solicits offers by issuing a Request for Proposal (RFP). The contractor's response constitutes an offer, and the subsequent contract award constitutes acceptance by the Air Force for contractor development of the system. Air Force Regulation 70-15, "Source Selection Policy and Procedures," provides the methods used in

competitive procurements. A Source Selection Plan (SSP) prepared by the SPO is submitted to the Source Selection Advisory Council (SSAC). The SSAC, chaired by the Source Selection Authority (SSA), evaluates the RFPs, approves the SSP, selects the source, and announces the contract award. Source Selection authority is normally delegated to the AFSC Product Division.

The SPO is the official AF organization established to acquire a system within cost, schedule, and performance criteria established. The program manager is responsible for all technical and business decisions relating to the system acquisition. Usually within a SPO organization there exists: 1) Program Control office responsible for planning and financial matters; 2) Configuration Management office responsible for formalizing system requirements into specifications and controlling the system configuration; 3) Engineering division which provides technical direction to the contractor and assures compatibility of all system elements; 4) Procurement organization responsible for procurement activities, and 5) Production Management office responsible for production activities.

All systems usually proceed through a five-step acquisition life cycle. The Conceptual Phase entails the identification and exploration of alternatives. The Validation Phase is used to refine solutions through study and analysis, and prototype testing and evaluation. The Full Scale Development Phase is where the principle items of the system, including

support equipment, are designed and fabricated. The Production Phase covers the period when systems are being built and fielded. Finally, the Deployment Phase is that period when equipment is provided to and used by operational units.

From initiation of the Conceptual Phase until completion of Production, the system is managed in accordance with AFR 800-2 policies and procedures. Systems engineering is controlled by AFR 800-3, "Engineering for Defense Systems," and MIL-STD 499A, "Engineering Management," policies and procedures. Configuration Management is conducted in accordance with MIL-STD 483, Configuration Management Practices for Systems, Equipment, Munitions and Computer Programs, procedures which establish the "baseline management" concept. Documentation of system development and production specifications (Part I and Part II specifications) is in accordance with MIL-STD 490, "Specifications Practices." Technical reviews and audits are conducted periodically to insure contractual compliance. These reviews and audit policies and procedures are lineated in MIL-STD 1521, "Technical Reviews and Audits for Systems, Equipment, and Computer Programs." AFR 800-14, "Acquisition and Support Procedures for Computer Resources in Systems," is a relatively new procedures document which provides detailed information to the SPO on the acquisition of computer resources which may be embedded in the system (Ref 1).

This has been a brief look at the process by which weapon systems are normally acquired in the Air Force, with specific attention to those elements which are directly related

to computer resources. A relatively new DOD Directive, 5000.29, "Management of Computer Resources in Major Defense System," (April 1976) has given added top-level emphasis to the area of embedded computer resources, but it is questionable whether it was able to exert any impact on many of the major systems in development prior to its publication. The rapid growth in computer utilization in the weapon system area and the technological evolution accompanying that growth must be recognized and dealt with at every management level. It is estimated that in the very near future the resources involved will reach a total of 40,000 computers and 110,000 computer programs in the Air Force alone (Ref 22:55).

General Purpose Computer Resource Acquisition

General Purpose Computers are usually identified as those which are off-the-shelf, commercially available, automatic data processing components, regardless of use, size, quantity, or price. They are designed to be applied to the solution or processing of a variety of problems, not for specific application (Ref 17:1).

Air Force management of General Purpose Computer resources are rooted in: 1) the Federal Legislation Public Law 89-306, October 1965 (the Brooks Bill), which regulates the acquisition of computer equipment, supplies, and services; 2) the Bureau of the Budget Circular A-71, March 1965, which delineated responsibilities for certain ADP acquisition functions to Government Services Agency (GSA); and 3) the Federal Property Management Regulations, and Federal Information Processing Standards

Publications (FIPS PUBS) (Ref 36:60). DOD Directive 5100.40, "Responsibility for the Administration of the DOD Automatic Data Processing Program," and DOD 4105.55, "Selection and Acquisition of Automatic Data Processing Resources," are the primary general purpose computer policy directives. Air Force Regulation 300-12, "Procedures for Managing Automatic Data Processing Systems Documentation, Development, Acquisition, and Implementation," is the basic Air Force document which interprets and implements the DOD directives.

The Assistant Secretary of the Air Force for Financial Management is designated as the senior ADP official. In accordance with the 300 series AF Regulations, authority to manage the Air Force ADP program is delegated to the Director of Data Automation under the direction of the Comptroller of the Air Force. Command ADP Program Single Managers exist at each of the major commands and are responsible for management of all the ADP programs within their organizations. The HQUSAF Data Automation Panel is responsible for review and approval authority for acquisition of general purpose computer equipment within designated financial thresholds (Ref 4).

Under this management structure, requests for new ADP resources are submitted in the form of a Data Automation Requirement (DAR) to the designated approval authority (depending on the financial threshold). Upon approval of the requirement, a Data Project Directive (DPD) is issued to the developing or acquisition agency, which, in turn, prepares a Data Project Plan (DPP). The DPD grants approval, assigns responsibilities,

and authorizes resource expenditures. The DPP describes actions to be taken by the development agency to achieve project performance, schedule, and cost objectives.

Budgeting and funding of the Air Force ADP Program are accomplished through the ADP Management Information System (ADPMIS) (RCS:DD-COMP(AR)996), which tracks progression from early functional requirement identification through Air Staff validation and entry into the Five-Year Defense Program (FYDP). Major cost incurrences are in the 3080 (Other Procurement) and 3400 (Operations and Maintenance Appropriations).

Most general purpose systems cannot be procured without prior approval of the GSA. An Agency Procurement Request (APR) is submitted to GSA, which can conduct the procurement or issue a Delegation of Procurement Authority (DPA) to the Air Force. AFR 70-15 source selection policies are applicable to the acquisition of general purpose computers.

The general purpose acquisition management philosophy, as presented in AFR 300-12, does include a five-phase life cycle development scheme, specific documentation standards, and an established sequence of reviews and audits. However, much of the terminology in this system does not match that in the embedded computer acquisition scheme and, of particular concern, is the entirely different documentation method. General purpose systems are documented in accordance with DOD Standard 7935.15, "Automated Data Systems Documentation Standards."

The "300 Series" management structure in its present form puts the ADP management body in the simultaneous positions

of advocate and adversary. That is, they are charged with holding down costs, scoping functional requirements, and enforcing utilization policy while, at the same time, being responsible for approval, acquisition, and implementation of general purpose computer systems in the Air Force.

Software Cost Estimation

As indicated by the Deputy Assistance Secretary of Defense for Material Acquisition in the October 1975 Defense Management Journal:

> The most critical issue facing DOD is the increasing use of and dependence on software in weapons systems without the proven management and production methods necessary to control its direct and indirect costs. Life cycle costing must be fully applied. The primary objective is to make top-level DOD management aware of the impact of software on the costs of weapon systems [Ref 19:1].

The 1975 DOD Weapon System Software Acquisition and Management Study conducted by Mitre Corporation concluded that:

> Meaningful cost information was not generally available. This was apparently due to lack of common definitions for the components of software costs, to regulations not requiring software to be broken out and maintained separately from hardware, and to a lack of detailed historical cost records. It was also noted that cost information was rarely correlated with technical information for management purposes. Future efforts to determine the cost of software in weapon systems should include (start with) the development of a management cost model and agreement on its content [Ref 26:6].

The DOD Software Study concluded that:

> Formal definition, reporting, collection, analysis and feedback of software cost information would improve managements visibility of software. It would provide information in the future so that major areas could be identified where DOD software costs are occurring and thus identify areas for possible improvements in cost and performance [Ref 26:7].

Many software acquisitions have experienced cost and schedule overruns leading to unanticipated cost growth. These experiences have highlighted the need to improve methods of software cost estimation. Software cost estimation is essential to budgeting, allocation of resources, and control of expenditures throughout the life cycle of a system. Accurate predictions of software costs are required in order to make practical and realistic tradeoffs between system capabilities and life cycle costs.

The central problem is that software estimation is difficult and extremely error prone. One of the most grievous problems with software cost estimates is that it is often difficult to determine until very late in the development process just how wrong they are. Two of the fundamental reasons for poor software cost estimates are:

1. the high risk and uncertainty involved in software developments
2. The lack of a quantitative data base of cost measurements on which to base cost estimates

Cost estimates may be prepared at any point in the acquisition cycle, but it is important to have estimates at the program decision point (between the Conceptual and Validation Phases), at the ratification decision point (between Validation and Full-Scale Development Phases), and at the production decision point (between Full-Scale Development and Production Phases). Because of the difficulty of accurately estimating software development costs, especially at points in the acquisition life

cycle where adequate technical information is not yet available, an iterative cost estimation process is the only way to obtain reasonably valid cost estimates.

A number of cost estimating techniques are used within the software industry. They are often referred to by different names and are sometimes used in combination. The most promising technique is parametric modeling. The parametric technique involves the identification of cost variables and quantification of their relationship to cost. Any new cost estimate can be made by estimating values for the cost variables and then computing the cost using the equations which express the cost estimating relationships. Empirical data is used as an objective reference. Information used in the estimation process normally includes: 1) allocation of requirements to software modules; 2) estimates of number of object instructions per module; 3) complexity and technological risk; 4) computer of choice; 5) higher order language of choice; 6) type of software to be developed; 7) technical experience of the developer; 8) length of development time; 9) performance record in number of instructions and development man-months; and 10) management factors to do with productivity rates, error rates, and availability of computer time (Ref 10).

Two recent theses in the Department of Systems Management (Devenny, T.J., GSM76S and Schneider, GSM-77S) both addressed the problem of Air Force software cost estimation. One thesis concentrated on efforts in the command and control software area through an analysis of activities at the Systems Command

Electronic Systems Division (ESD) The other thesis addressed Avionics software cost estimation activities at the Aeronautical Systems Division (ASD). Both of the theses in question made recommendations regarding the adoption of a "common" software cost estimation technique. Both theses also made recommendations regarding further study of a promising new software cost estimation system known as the RCA PRICE-S model. Neither thesis had sufficient time nor adequate cost data to validate the model sufficiently to make recommendations regarding its adoption as a standard cost estimation methodology.

Research Effort

Objectives. Questions to be considered in conducting this research were the following:

1. Are there differences in software development costs other than inherent complexities of machine and language dependencies which would negate the possibility of adopting a standard for software cost estimation?

2. Does a specific software cost estimation model such as the RCA PRICE-S model have universal applicability for Air Force embedded, command and control, and management data systems, and could the single model gain acceptance as a standard for cost estimation and reporting purposes?

Scope. While there are a number of software cost estimation techniques available and in use today, this study restricted attention to RCA PRICE-S, one of the more promising automated systems presently in use by government and industry.

Data gathering was limited to the Aeronautical Systems Division (ASD), Electronic Systems Division (ESD), and the Air Force Data Systems Design Center (AFDSDC), which are the three organizations responsible for the major portion of Air Force software development and acquisition. Data gathering was limited to a cross-section of past programs (for which cost information is available) in the three major areas of embedded, command and control, and management data systems.

Methodology. Data collection for this research was divided into three phases. First, a general literature review on current software cost estimating state-of-the-art techniques was conducted. This phase provided the researcher with the necessary background and information for evaluating the specific methodology to be examined.

Second, information on the cost estimating system utilized in the analysis was gathered. This phase was necessary to familiarize the researcher with the requirements for operation of the systems.

Finally, the researcher gathered historical data on previous Air Force software development projects. Data was gathered through personal interviews with personnel at the major development centers.

Data Analysis. Historical data from past development efforts were collected based on the cost estimating requirements identified in the PRICE-S system along with data on the actual system costs. The historical data were utilized as input data to the cost estimating systems in question. Comparisons of the

system generated cost estimates with the actual historical costs was conducted. Analysis was conducted to compare the outputs of the test system to actual historical cost to determine if the predictions generate statistically significant differences in software development costs.

The purpose of this chapter was to provide a basic understanding of the computer resource acquisition process and define the scope of this research effort. The following chapter will expand on the basic acquisition process and include a general discussion of the software development management process. An understanding of these two processes is essential for anyone involved in estimating the cost of developing computer software.

II. Background

The history of the software industry has been marked by cost overruns, late deliveries, poor reliability, and user dissatisfaction. While these problems are not unique to computing, the record seems to indicate that software developers as a group are less successful in meeting quality, cost, and schedule objectives that their hardware counterparts.

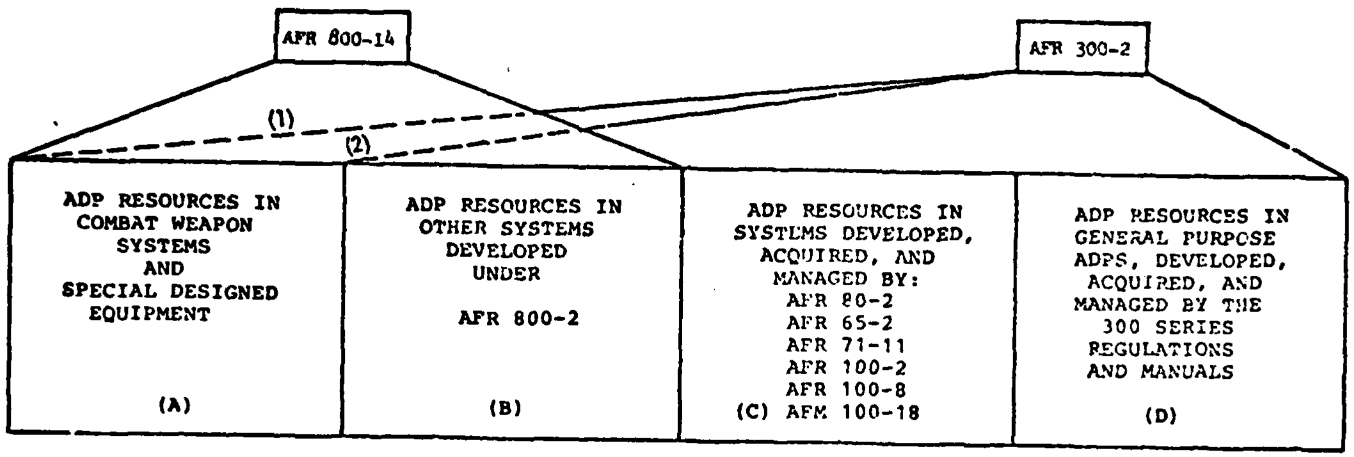
Recent advances in the state-of-the-art in computer software development techniques hold some promise for reducing Air Force expenditures. Some of these techniques, referred to as modern programming practices, are now being applied by various DOD and industrial software developers. Some of these practices include project planning and organization techniques, design methodologies, coding and testing practices, use of programming support tools, documentation standards, configuration management and change control techniques, and the procedures and guidelines necessary to employ these practices in a disciplined manner. The Air Force is encouraging contractors, as well as its own development organizations, to employ beneficial practices in developing software. Those practices found to be effective in reducing costs are being standardized into specifications and guidelines.

Software Management

In general, two types of acquisitions involving software have been distinguished to which two basic series of regulations can apply: 1) ADP procurement, management, and development regulared by the AFR 300 series; and 2) acquisition of major systems, including embedded computer resources, regulated by the AFR 800 series. The two series are not mutually exclusive. The Air Force ADP Program Single Manager established by AFR 300-2 is responsible for providing ADP technical and managerial expertise to AFR 800 series acquisition programs through HQ USAF coordination. Also for 800 series programs, several specific regulations in the 300 series can be employed. Figure 1 (Ref 3:19) illustrates the major relationships between the two series.

Embedded Systems. One of the major features associated with software development is the notion of a life cycle concept as previously discussed. This life cycle concept views the software as going through a series of phases. Depicted in Figure 2 (Ref 24:6.50) is a typical software development and configuration management approach which conforms to the general policies and guidelines established by the DOD for embedded computer systems. The key concept in configuration management is the "baseline" which is established by customer review and acceptance of a baseline specification document. Baselines are so called because they are the bases, or reference points, for subsequent development and control. The configuration management approach pictured in Figure 2 and

APPLICATION OF POLICY GUIDANCE TO MANAGEMENT OF ADP AND COMPUTER RESOURCES



Category A - These items are excluded from the DOD and Air Force ADP programs. They are subject to the policies of the AFR 800-14.

Category B - The ADP resources integral to these systems are subject to policies of AFRs 800-14 and 300-2.

Category C - Management of the ADP resources in these systems are subject to AFR 300-2 and the cited regulations. As these ADP resources are dedicated to the systems they support, primary management stems from the basic regulation governing the system.

Category D - These systems are developed, acquired, operated, and managed using the AF 300 series of directives.

(1) Applicability of AFR 300-2 to Category A is limited to technical and managerial expertise, to be provided by ADP Program Single Manager organizations by means of review, consultation, recommendation, and HQ USAF coordination.

(2) The PMD specifies the pertinent AFR 300-2 policy requirements.

Fig 1. AF 800 vs. 300 Regulation Relationship [Ref 3:19]

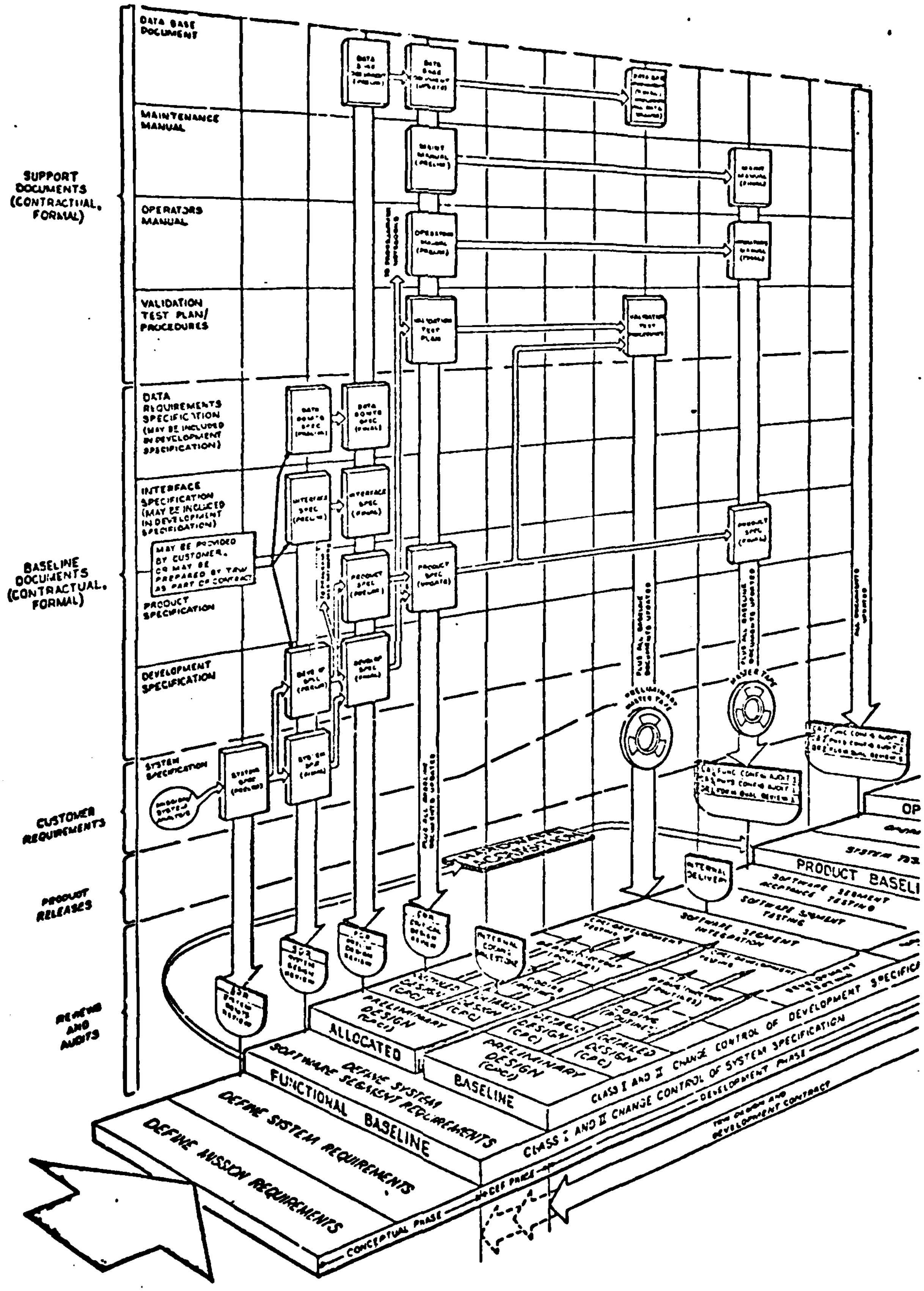


Fig 2. Embedded Software Life Cycle Management Model [24:6.50]

briefly summarized in Figure 3 (Ref 1: 9) includes the life cycle phases, the baselines, the baseline specifications and documents, the computer program hierarchy, and the management control vehicles.

The Conceptual Phase of the software life cycle is initiated by an analysis of mission requirements through a definition of operational concepts, environments, and constraints. System feasibility studies are normally conducted to formulate the basic system requirements and determine technological and economic feasibility. Systems Engineering efforts during this phase include the allocation of functions between man and machine, and the determination of computer performance characteristics. Overall system performance and testing requirements are defined and major system elements and interfaces are established. The major output of the Conceptual Phase is the System Specification document. Acceptance of the System Specification by the customer during the System Requirements Review (SRR) signifies the establishment of the Functional Baseline (Ref 31:2-10).

The Definition (Validation) Phase is started by defining the interface requirements between the operational functions and includes an initial allocation of performance requirements by segment, the development of a schedule, and the establishment of control techniques. The tasks to be performed by the individual software programs are defined at this time, as well as the manual tasks and procedures required for operation of the equipment and the automated tasks to be

System Acquisition Life Cycle Phase	Primary Software Product Documents**			
	Document Type	Time of Issue	Originator	Governing RSS
CONCEPTUAL PHASE Purpose: To define overall mission and system requirements.	Preliminary System Spec (or Preliminary System Segment Spec)	At SRR	Program Office or conceptual phase contractor	MIL-STD-490, MIL-STD-483, and appropriate DIDs.
VALIDATION PHASE Purpose: To validate system concepts and establish the functional requirements for major end items of the system.	1. Final System Spec (or Final System Segment Spec) 2. Preliminary CPCI Development Specs	At SDR	Validation phase contractor	MIL-STD-490, MIL-STD-483, and appropriate DIDs.
FULL-SCALE ENGINEERING DEVELOPMENT PHASE Purpose: To design, build, and test system end items; to integrate end items into a complete system; and to test system under as nearly operational conditions as possible.	1. Final CPCI Development Specs 2. Preliminary CPCI Code-To Product Specs	At PDR	Software development contractor	MIL-STD-490, MIL-STD-483, and appropriate DIDs.
	Final CPCI Code-To Product Specs	At CDR	Software development contractor	MIL-STD-490, MIL-STD-483, and appropriate DIDs.
	--	--	--	--
	CPCI As-Coded Product Specs	At PCA	Software development contractor	MIL-STD-490, MIL-STD-483, and appropriate DIDs.
	1. User Manual 2. Positional Handbooks 3. Computer Programming Manual	At Product Baseline	Software or hardware development contractor or system integration contractor, as appropriate	Appropriate DIDs.
PRODUCTION/ DEPLOYMENT AND OPERATION/ MAINTENANCE PHASES Purpose: To field system to operational sites and install and test them, then to operate and maintain them.	--	--	--	--

Fig 3. Embedded Software Life Cycle Management Summary [Ref 1:9]

Fig 3, continued

System Acquisition Life Cycle Phase	Software Development Tasks	Reviews and Audits	Baselines
CONCEPTUAL PHASE Purpose: To define overall mission and system requirements.	Preliminary statement of software requirements, if available.	1. System Requirements Review (SRR)[a] 2. DSARC I (Program Decision)	Functional Baseline (configuration control of Preliminary System Spec or Preliminary System Segment Spec)
VALIDATION PHASE Purpose: To validate system concepts and establish the functional requirements for major end items of the system.	Major system characteristics are refined through studies, system engineering, and preliminary equipment and computer program development, test, and evaluation.	1. System Design Review (SDR) 2. DSARC II (Ratification Decision)	Allocated Baseline (configuration control of System Spec or System Segment Spec and usually of CPCI Development Specs)
FULL-SCALE ENGINEERING DEVELOPMENT PHASE Purpose: To design, build, and test system end items; to integrate end items into a complete system; and to test system under as nearly operational conditions as possible.	1. Preliminary Design. Definition of the CPCIs in terms of functions, external and internal interfaces, storage allocation, operating sequences, and data base design.	Preliminary Design Review (PDR)	--
	2. Detailed Design. Definition of CPCI structure, interface logic, and data base to point where coding can begin.	Critical Design Review (CDR)	--
	3. Coding and Unit Test. Routines and data files are coded, debugged (will compile), and checked out (will produce correct results from predefined inputs).	Test Readiness Review (TRR, a contractor internal review)	--
	4. Integration and Test		
	a. CPCI Tests. CPCIs are tested together in increasingly larger combinations until all CPCIs developed by the same contractor are functioning together correctly.	1. Functional Configuration Audit (FCA) 2. Physical Configuration Audit (PCA)	Preliminary Product Baseline (configuration control of System Spec or System Segment Spec, of CPCI and CI Development and Product Specs, and of CPCIs and CIs themselves)
	b. Integrated System Testing. All CPCIs and hardware CIs of the system are tested together to verify that the system meets the requirements of the system spec.	1. Same as preceding (FCA, PCA), as required 2. Formal Qualification Review (FQR) 3. DSARC III (Product Decision)	Product Baseline (configuration control of same items as for Preliminary Product Baseline, but updated)
PRODUCTION/ DEPLOYMENT AND OPERATION/ MAINTENANCE PHASES Purpose: To field system to operational sites and install and test them, then to operate and maintain them.	Installation, maintenance, and modification, as required.	--	None (usually continuing configuration control of specs and products)

performed by the computer. General requirements for the design, development, test, and validation of the software are specified during this phase. The Allocated Baseline is established through customer acceptance of the software Development Specification at the System Design Review (SDR) (Ref 31:2-13).

A major element of the Development Phase of the life cycle is the allocation of inputs, outputs, and functions to various system elements, the segmentation of programming tasks into specific packages and the development of a functional flow. A Preliminary Design Review (PDR) is conducted early in the phase in order to confirm the design integrity of the proposed system. Charts, diagrams and descriptions are prepared for each software item in sufficient detail for eventual coding. At this point, a Critical Design Review (CDR) is conducted for the purpose of insuring that the Product Specification, containing the actual design, meets the development requirements. Following the CDR, the actual coding of the computer programs occurs. Developmental testing by the individual programmers occurs concurrently with the coding activities. Functional and Physical Configuration Audits (FCA, PCA) and Formal Qualification Review (FQR) are conducted on each software program to insure acceptance of the software and documentation. Finally, system and operational tests of the entire hardware and software system are conducted to insure that performance and design requirements contained in the specifications are met (Ref 31:2-16).

The Operational Phase involves turnover of programs and documentation to the operational user for subsequent operation, maintenance, and refinement.

Because the actual software program itself is invisible to the user, the development of an adequate documentational representation is essential. The baseline documents are normally deliverable items to the customer. The first major document is the System Specification which details the mission requirements of the system, allocated functional requirements to the individual configuration items (programs), and defines the configuration item interfaces required. The Development Specifications, which establishes the Allocated Baseline, describes in detail all of the requirements necessary to design and test the individual programs. The Product Specification, another of the major documents in the series, establishes the Product Baseline. This document is the technical description of the program and will include an actual listing of program code. Other documents in the series include Test Plans, Operator's Manuals, and Program Maintenance Manuals (Ref 31:4-6).

A series of progressively more detailed reviews and audits are scheduled at various decision points in the life cycle to allow the program manager to assess progress and establish new baselines for each of the individual software programs. The System Requirements Review (SRR) is the first in the series, and its purpose is to review the System Specification in order to determine if the preliminary requirements allocation satisfied mission requirements. The

System Design Review (SDR) is conducted to insure that the allocated functional requirements in the preliminary Design Specification fulfill the System Specification Requirements. A Preliminary Design Review (PDR) is usually conducted for each program to evaluate the basic design approach for completeness, adequacy, and compatability with the allocated requirements in the Development Specification. The last in the systems engineering oriented reviews is the CDR (Critical Design Review). The CDR is conducted prior to actual coding to insure that the detailed design solution in the Product Specification meets the performance requirements contained in the Development Specification. Finally, the Functional Configuration Audit (FCA) is conducted to insure that actual program performance is in compliance with the Development Specification; the Physical Configuration Audit (PCA) verifies that the final program is as described in the Product Specification; and the Formal Qualification Review denotes contractual acceptance of the program by the customer (Ref 31:8-5).

The configuration management system depicted here has evolved over time, largely as a response to increased demand for improved software development techniques. It has also been suggested that cost predictions cannot be fulfilled unless the mechanism for management control is satisfied in advance. The major emphasis of the system is to produce accurate software documentation which becomes the instrument by which management controls the project. Technical reviews are conducted against predetermined criteria for the purpose

of establishing necessary baselines. Configuration management controls and procedures are applied to assure that changes are implemented and tested properly. The system provides a data reporting and control system to assure that all software configuration data are analyzed, reported, and available when needed (Ref 35:2).

General purpose. Unlike embedded computer software which is normally developed by contractor personnel (as part of a larger weapon system) under the guidance and direction of DOD program management personnel, general purpose systems are usually management information oriented systems and are developed by the various Air Force agencies involved (e.g., Personnel, Accounting and Finance). The software project management concept shown in Figure 4 (Ref 5:2-2a), which is analogous to that of the embedded system, has been developed for general purpose software. The concepts of life cycle phases, documented baselines, and decision points keyed to specified management reviews are carried through this system almost intact. Effective software project management is still the key to development success.

The first step in the acquisition of an ADP capability to fulfill a mission or operational requirement is a user analysis of need, identification of alternatives, and documentation of requirements. This conceptual requirements document is known as the Data Automation Requirement (DAR). The system development process is initiated by a Conceptual Phase during which the user determination of mission

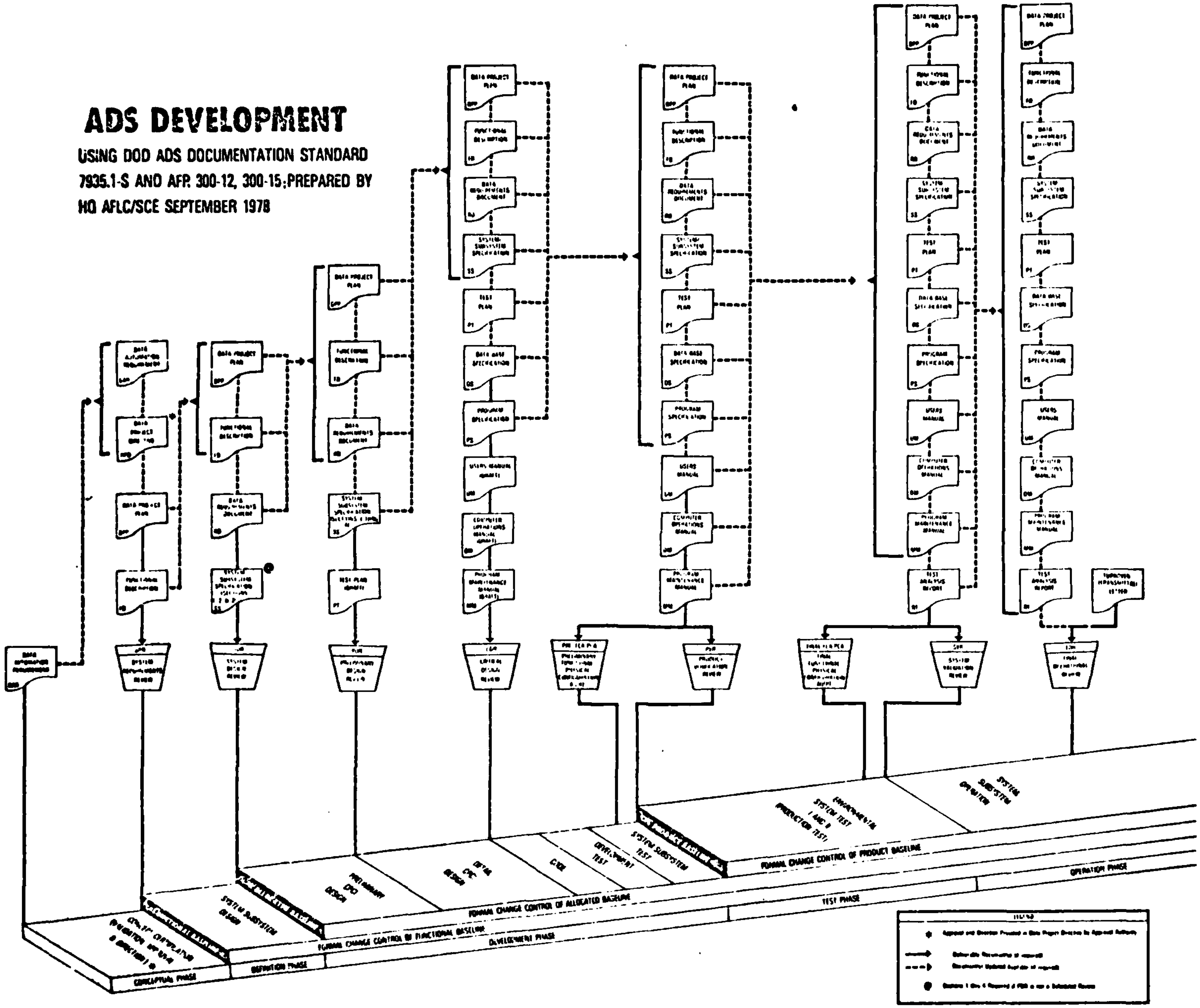


Fig 4. General Purpose Software Life Cycle Management Model [Ref 5:2-2a)

requirements and system requirements is completed. The Definition Phase encompasses the development of the functional requirements of the major software and hardware elements of the system. During the Development Phase, the preliminary analysis and design, detailed analysis and design, development (coding, debugging, and checkout), and testing of system elements is accomplished. The Integration Phase includes installation, integration and testing of system elements in the operational environment. Operation, maintenance and product improvement are the major activities in the Operational Phase of the life cycle (Ref 5:2-2).

The characteristics of an evolving system and its configuration items are defined and documented in increasing detail at logical transition points or baselines in a manner similar to that of the embedded computer software. The Functional Baseline marks the end of the Conceptual Phase and is established by a Functional Description (FD) document. The Allocated Baseline established by the System Specification (SS) marks the end of the Definition Phase. The Product Baseline defines the end of the Development Phase and is established by user acceptance of the Program Specification (PS).

The documentation scheme for the general purpose systems is again similar to the embedded systems. The Functional Description (FD) is a document which states the mission requirements for a system, allocates requirements to functional areas for configuration items and defines the interfaces between or among the configuration items. The System

Specification (SS) is a technical document that governs the development and testing of a computer program. The SS defines performance, interface, and other technical requirements in sufficient detail to permit design, coding, and evaluation. The Program Specification (PS) is the document which defines the characteristics of the computer program in sufficient detail to permit coding. The final version of the PS is utilized in performing program maintenance and modification. Additional test and evaluation, data base, and operational support documents are developed to assist in the configuration management activities.

Again, a series of reviews and audits are scheduled at meaningful points during the development cycle to assess progress and establish configuration identifications. The specific number, content, and conduct of the reviews and audits are normally included in the governing documentation and established by agreement between the user and the development agency. The reviews and audits are divided into two basic types, those which are primarily system engineering oriented, and those which are configuration management oriented. The System Requirements Review (SRR), which is the first of the engineering oriented reviews, is conducted to review the users requirements and determine details for the development of an FD. A System Design Review (SDR) is conducted to insure the adequacy of the FD in satisfying mission requirements and to evaluate the SS for technical understanding of requirements. The Preliminary Design Review (PDR) is a technical

evaluation of the basic design approach for the computer program. Finally, a Critical Design Review (CDR) is conducted to insure that the detailed design solution reflected in the PS satisfies the performance requirements established in the SS. The Functional Configuration Audit (FCA), Physical Configuration Audit (PCA), and the Formal Qualification Review (FQR) are configuration management oriented activities conducted to verify that actual performance is in compliance with the SS, and that the coded version of the program conforms to the technical documentation description (PS) (Ref 5:5-1).

The software project management techniques portrayed here, both for the embedded as well as the general purpose computer software, describe the activities encompassing the planning, control, engineering, and supervision involved in producing an end product--an operational computer program. The approaches to the management of a project depicted here facilitate the orderly analysis, assimilation, and resolution to the problems of complex development efforts. The concepts and principles of project development herein are the key elements in evolving an adequate software cost estimation methodology. Each phase has structured inputs and outputs which can be evaluated to determine cost elements. The phases provide guidelines for managing the development process and uniform output so that quantitative measures of results can be obtained.

Software Cost Estimating Factors

As previously pointed out, the cost of developing software

is rising rapidly and will soon become the most expensive element in any computer project. Because of this, more emphasis within DOD is being placed on estimating the cost of software. As a result of this emphasis, many studies have been conducted addressing the software cost estimation area. One recent study concluded that:

> The objective of software cost estimation is to determine what the costs of software will be. A good cost estimate must also show when and how the costs will be incurred. Such information not only provides justification for a cost estimate, but is also essential to budgeting, allocation of resources, and control of software expenditures throughout the life cycle of a system. . . . Many software acquisitions have experienced cost and schedule overruns which have led to unanticipated cost growth for the software and for the overall system. These experiences have highlighted the need to improve our methods of software cost estimation [Ref 10:5].

The same study also concluded:

> There is only one real problem with software cost estimation: overruns. Unexpected costs due to poor cost estimation are harder to accept than planned costs. On the basis of a cost estimate too many decisions are made which cannot be undone when the cost estimate is proven wrong. One of the most grievous problems with software cost estimates is that it is often difficult to determine, until very late in the development process, just how wrong they are [Ref 10:9].

Software cost estimation is best accomplished through an iterative process. As the program advances in time and detail (i.e., system requirements are defined, alternatives are studied and a feasible approach is developed), additional and more definitive information becomes available which can be utilized in obtaining realistic cost estimates. The process of estimating software development resource and time requirements is a complex task requiring in-depth

knowledge of each program. Cost estimates may be prepared at any point in the acquisition cycle, but it is important to have an estimate at the major decision points:

1) Program Decision (between Conceptual and Validation Phases)
2) Ratification Decision (between Validation and Full-Scale Development Phases)
3) Production Decision (between Full-Scale Development and Production Phases)

Many studies and analyses have been conducted in an effort to determine what factors impact software development schedules and costs. The following is a brief summary of some of the major factors thought to impact the software development process, including both cost and schedule. The complexity of the software being developed is one of the most important factors effecting cost and schedule. Complexity has a direct correlation to programmer productivity (measured in output per unit of time). Programmer productivity varies significantly with the type of development job. Because the relationship between programmer productivity and complexity varies due to the creative nature of the task, the attributes of the software problem, unique individual differences, and the variability of terms in measuring output, the measurement remains largely a subjective assessment. Basically, the derivation of a software complexity index involves the determination of the applications characteristics. Some complex applications may involve innovative or high risk technology (Ref 21:36).

One recent study concluded that:

1) embedded system productivity rates for avionics software were lower for onboard flight programs than for automatic test equipment or related simulators;

2) command and control software could expect a 40 percent decrease in normal productivity because of the complexity in programming real-time requirements;

3) business applications show the highest levels of productivity;

4) scientific applications should use lower productivity rates because of the use of complex computational algorithms (Ref 25).

Another important factor effecting the cost of software is the size of the development effort. Estimating the size of software programs has proven the greatest source of error in analyses to project resource requirements of software development. Use has been made of both object instructions (output of the compiler) and source instructions (output of the programmer) for measurement of program size. Estimates are generally given in object instructions. The rapid expansion in the use of High Order Languages (HOL) has complicated efforts in developing adequate cost relationships for software size. The estimated number of instructions used may include software that must be developed but not delivered (especially true for embedded systems support software). It is commonly accepted that the size of the software, however measured, is related linearly to cost (Ref 25:39).

The types of requirements specified for the system can impact the allocation of resources to the development project. The completeness, complexity, and compatibility of performance requirements will have a direct effect on development costs and schedules. Special display equipment, real-time operations with critical response times CPU memory size and time constraints, and concurrent development of software and hardware components have all been shown to reduce productivity. The quality of performance requirements specifications can impact the development process. Too little detail allows for ambiguities in interpretation, while highly detailed performance requirements will invariably include some specification of design ultimately limiting development alternatives.

Documentation requirements for a system acquisition can be very costly. The cost of documentation can include not only that relating to the specific design approach, but also that relating to configuration management, program control, and technical progress documentation. A recent Government/Industry Software Sizing and Costing Workshop indicated that documentation costs approximately 10 percent of the total software development or $35-$100 per page, depending upon the amount and complexity of the analysis required in document production (Ref 2).

Software quality attributes, which relate to the products required capabilities and performance characteristics, can have a direct effect on project cost and schedule. Quality attributes such as maintainability and reliability are normally

specified for embedded systems in terms of performance requirements. Conflicting attributes such as modularity and efficiency can cause a requirement for less than optimum design decisions. While the effect of quality requirements may be an increase in development costs, they may ultimately lower the cost of maintenance and support activities. These types of quality requirements have proven difficult to quantify in the current state of software technology (Ref 25:42).

The software development schedule, or total amount of calendar time allocated to the project, has a significant impact on costs. Generally, the development schedule is a fixed constant. Because the development tasks are largely sequential in nature, they cannot arbitrarily be compressed or reorganized within the allocated schedule. Therefore, the number and sequence of tasks to be performed in a given time period will indicate the manpower required. There appears to be a definite relationship between program size and development time. Management cannot diminish the development time of a system without increasing the difficulty. An optimum man-loading appears to exist, loading above or below which will negatively impact costs and schedules. The manner of allocation to specific activities is also important. Too little time and effort spent in analysis and design will have enormous impact on eventual costs to correct design deficiencies. As development progresses, it becomes more and more costly to resolve design errors.

Software development projects do not always involve the

generation of an entirely new code but utilize some portion of the existing code which must be transferred or retrofitted. Costing software retrofits must include analysis of the existing system, decomposition of the retrofit requirements and estimation of the costs of modifying the existing programs to interface with new software. Transferring an operational software system to new equipment can also require detailed analysis of the software in light of equipment operational differences.

Software development requires personnel who possess both analytical as well as creative skills in solving complex problems. Cost estimation usually involves the derivation of a productivity figure per manpower unit for a person with an average skill level. The costs added to a late project by adding additional manpower may be more than those incurred by the additional manpower cost. There may also be further costs resulting from the additional training and coordination required. Some development projects have shown that the increased complexity in the development process caused by additional manpower have caused the project to fall further behind schedule. As much as 20 percent of any manpower requirement for a particular project may be utilized in support activities not directly related to the production of code. Recent studies have shown variations in productivity rates for experienced programmers of 10 to 1. The use of application-suitable HOLs can have an impact on productivity averages. Programmer productivity increases by as much as

a factor of five have been experienced with the use of a HOL (Ref 27:46).

The vast majority of software cost estimates are derived from the basic sizing parameter of estimated number of instructions, derived using historical experience or engineering judgment and then applying various other factors to this parameter to determine cost. Variations of this approach go from simple "rules of thumb" to complex mathematical models. It is not surprising that the factors that affect software costs are complex and their quantification difficult. In order to arrive at an accurate cost estimate, it is necessary to take some or all of these factors into consideration.

RCA PRICE Software Model

A number of cost estimating techniques are used within the software industry today. One of the largest and fastest growing techniques being used is parametric modeling. The parametric models for estimating the cost of software development consist of an equation, or group of equations, which express a quantifiable relationship of a software project's cost to a number of cost variables. Derivation of the relationship of the cost to the variables is dependent upon analysis of historical and project variables. Based upon the quantified cost/cost variable relationships, new system estimates can be made by estimating the cost variables for the new system and inserting these values into the parametric model. The major advantages of parametric models

is tnat they are often computerized and require little or no software development experience for the user.

The RCA PRICE S (Programmed Review of Information for Costing and Evaluation) software model is one of the most promising of the parametric cost estimation models being used today. A number of the major U.S. corporations and DOD agencies involved in software development are presently using the model with good success. The model includes capabilities for calculating estimates for all programming applications including management data systems, command, control and communications, and embedded avionics. PRICE S provides for: interactive operational capability; an efficient problem description methodology using a small set of input factors; an internal self-checking mechanism for input data consistency; and a flexible feature allowing the user to tailor the model to organizational operating methods. The following PRICE S system description is extracted from course materials provided to the researcher during attendance at the RCA PRICE S training session conducted at Cherry Hill, New Jersey, 25-28 June 1979 (Ref 32).

Overview. System peripheral hardware configurations, system processor utilization factors, reliability requirements, economic factors and programming resource characteristics are incorporated as model inputs (Figure 5). The model provides standard cost and schedule estimates (Figure 6), as well as sensitivity analysis capabilities (Figure 7). Estimates are based on project size, type, complexity, and can incorporate

scheduling constraints when necessary. Cost summaries and resource expenditure profiles (Figures 8 and 9) are provided for each of the three major life cycle phases (Design, Implementation, Testing) for each of five cost categories (System Engineering, Programming, Configuration Control, Documentation, Program Management)(Figure 10). In cases where a user specified schedule is input, the system will compare this data with a typical industry schedule and provide appropriate cost adjustments for acceleration or stretch-out (Figure 11).

The PRICE-S system has three modes of operation: the Normal mode, the ECIRP mode, and the Design-to-Cost mode. The Normal mode is used to calculate costs directly from user inputs. The ECIRP mode allows the calculation of PRICE-S empirical factors from historical data by running the model essentially in reverse. If specific project data are input, the Design-to-Cost mode will allow the user to investigate the scope of possible alternative programs. The PRICE-S system involves, then, the evaluation of new requirements based on historical information through a few variables which can be adjusted for technological, economic, and organizational differences.

System Parameters. The PRICE-S system has been designed so that it can be operated with a small number of variable input parameters. The following discussion provides a description of the key variables of the model (Ref 32:Part IV, 1-50).

Application (APPL) is a single variable which, in effect, provides a measure of the program instruction mix. APPL

PRICE Software Model
Input Worksheet

Filename: ______

Page ___ of ___

Title ______________________

Application ______________________

Date __________

								Optional	
Descriptors	INST	APPL	RESO	FUNCT	STRU	LEVEL		INTEG	
Mix	MDAT	MONL	MREA	MINT	MMAT	MSTR	MOPR	MAPP	APPL8
New Design	DDAT	DONL	DREA	DINT	DMAT	DSTR	DOPR	DAPP	
New Code	CDAT	CONL	CREA	CINT	CMAT	CSTR	COPR	CAPP	
Device Types	TDAT	TONL	TREA	TINT					
Quantity	QDAT	QONL	QREA	QINT					
Schedule Data	CPLX	DSTART	DEND	ISTART	IEND	TSTART	TEND		
Supplementary Information	YEAR	ESC	TECIMP	MULT	PLTFM	UTIL		TARCST	

Notes:

GC 1610 8/77

RCA

Fig 5. PRICE-S Model Input Form (Ref 32)

```
                         --- PRICE SOFTWARE MODEL ---

                        DATE 11/10/77  TIME 08:41:23
MOBILE RADAR
                                                                  SAMPLE CASE

FILENAME: CLASS                    INPUT DATA
                                                               DATED: 07/22/77
DESCRIPTORS
  INSTRUCTIONS    36000         APPLICATION    0.0          RESOURCE       3.500
  FUNCTIONS           0         STRUCTURE      0.0          LEVEL          2.600

APPLICATION CATEGORIES          NEW DEVELOPMENT             SYSTEM CONFIGURATION
                   MIX          DESIGN         CODE         TYPES        QUANTITY
  DATA S/R         0.0          1.00           1.00         0              0
  ONLINE COMM      0.08         1.00           1.00         1              1
  REALTIME C&C     0.08         1.00           1.00         2              2
  INTERACTIVE      0.23         1.00           1.00         1              2
  MATHEMATICAL     0.28         0.50           0.70         ***            ***
  STRING MANIP     0.26         1.00           1.00         ***            ***
  OPR SYSTEMS      0.07         1.00           1.00         ***            ***

SCHEDULE
  COMPLEXITY       1.250
  DESIGN START     SEP 77       IMPL START      0           T&I START      0
  DESIGN END        0           IMPL END        0           T&I END        0

SUPPLEMENTAL INFORMATION
  YEAR             1977         ESCALATION     0.0          TECH IMP       1.00
  MULTIPLIER       1.000        PLATFORM       1.4          UTILIZATION    0.80

                                 PROGRAM COSTS

COST ELEMENTS                   DESIGN          IMPL          T & I        TOTAL
  SYSTEMS ENGINEERING            309.            14.           255.         578.
  PROGRAMMING                     40.            65.           104.         210.
  CONFIGURATION CONTROL           71.            19.           158.         248.
  DOCUMENTATION                   52.             6.            63.         121.
  PROGRAM MANAGEMENT              29.             6.            31.          66.
     TOTAL                       501.           110.           612.        1223.

                                ADDITIONAL DATA
DESCRIPTORS
  INSTRUCTIONS    36000         APPLICATION    5.299        RESOURCE       3.500
  FUNCTIONS         400         STRUCTURE      4.961        LEVEL          2.600

SCHEDULE
  COMPLEXITY       1.250
  DESIGN START     SEP 77       IMPL START     JAN 78       T&I START      MAY 78
  DESIGN END       JUL 78       IMPL END       NOV 78       T&I END        JUL 79

SEP 77                          SCHEDULE GRAPH
************ DESIGN ************                                           JUL 79
                  ************ IMPLEMENT ************
                                ****************** TEST & INTEGRATE ***************
```

Fig 6. PRICE-S Model Output Cost and Schedule Estimate (Ref 32)

--- PRICE SOFTWARE MODEL ---

DATE 01/17/78 TIME 11:02:34

SAMPLE CASE MOBILE RADAR

SENSITIVITY DATA

COMPLEXITY

RESOURCE		1.150		1.250		1.350	
3.400	COST	1085.	COST	1170.	COST	1260.	
	MONTHS	19.8	MONTHS	21.8	MONTHS	23.7	
3.500	COST	1132.	COST	1222.	COST	1315.	
	MONTHS	19.9	MONTHS	21.9	MONTHS	23.8	
3.600	COST	1180.	COST	1274.	COST	1371.	
	MONTHS	20.0	MONTHS	22.0	MONTHS	23.9	

Fig 7. PRICE-S Model Sensitivity Analysis Output (Ref 32)

--- PRICE SOFTWARE MODEL ---

DATE 04/24/78 TIME 11:14:37

SAMPLE CASE | MOBILE RADAR

MONTH	% COMPLETED DESIGN	% COMPLETED IMPL	% COMPLETED T & I	$ EXPENDED THIS MONTH	$ EXPENDED TOTAL	% EXPENDED THIS MONTH	% EXPENDED TOTAL
OCT 77	2.0	0.0	0.0	10.2	10.2	0.8	0.8
NOV 77	15.1	0.0	0.0	65.3	75.4	5.3	6.2
DEC 77	34.1	0.0	0.0	95.3	170.7	7.8	14.0
JAN 78	54.0	0.0	0.0	99.3	270.1	8.1	22.1
FEB 78	71.4	0.2	0.0	87.2	357.3	7.1	29.3
MAR 78	84.6	3.4	0.0	69.7	427.0	5.7	35.0
APR 78	93.3	11.8	0.0	52.8	479.8	4.3	39.3
MAY 78	98.0	25.7	0.0	38.5	518.4	3.2	42.4
JUN 78	99.7	43.3	0.0	28.2	546.6	2.3	44.7
JUL 78	100.0	62.0	0.3	23.6	570.2	1.9	46.7
AUG 78	100.0	78.7	1.3	24.5	594.7	2.0	48.7
SEP 78	100.0	91.2	3.4	27.0	621.6	2.2	50.9
OCT 78	100.0	98.0	7.2	30.7	652.3	2.5	53.4
NOV 78	100.0	100.0	13.0	37.1	689.4	3.0	56.4
DEC 78	100.0	100.0	20.8	48.1	737.6	3.9	60.4
JAN 79	100.0	100.0	30.8	61.2	798.8	5.0	65.4
FEB 79	100.0	100.0	42.8	72.9	871.6	6.0	71.4
MAR 79	100.0	100.0	56.1	81.3	952.9	6.7	78.0
APR 79	100.0	100.0	69.9	84.4	1037.4	6.9	84.9
MAY 79	100.0	100.0	82.9	79.8	1117.2	6.5	91.5
JUN 79	100.0	100.0	93.5	64.8	1182.1	5.3	96.8
JUL 79	100.0	100.0	99.5	36.4	1218.5	3.0	99.7
AUG 79	100.0	100.0	100.0	3.1	1221.6	0.3	100.0
ALPHA =	0.82	0.0	0.0	FOR PROFILE GRAPHS			
BETA =	0.18	1.00	0.18	RESPOND OK = 7			
PEAK/AV =	1.93	1.88	1.93				

Fig 8. PRICE-S Model Cost Expenditure Summary Output (Ref 32)

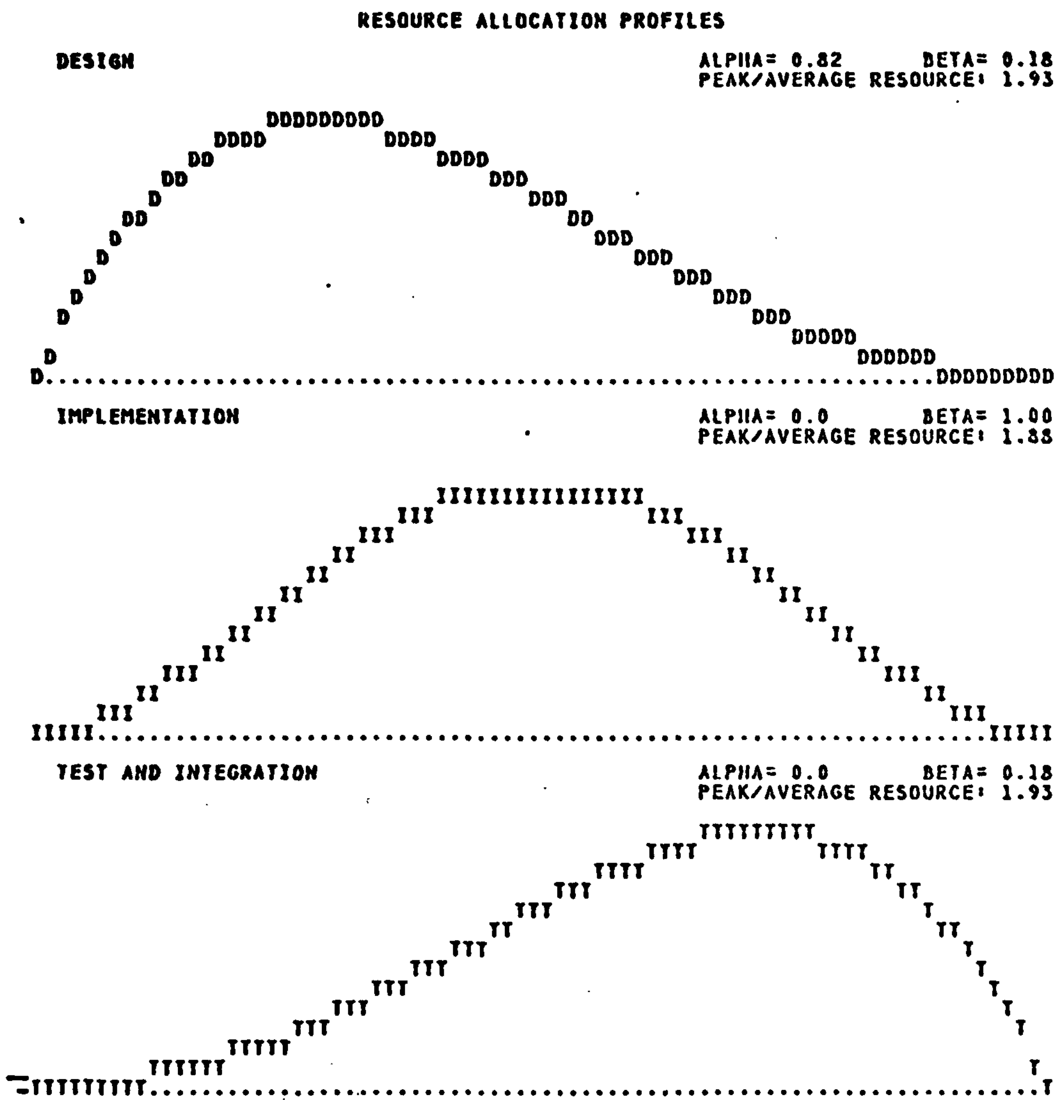


Fig 9. PRICE-S Standard Resource Expenditure Profile (Ref 32)

SOFTWARE PROGRAM COSTS
RIPPLE EFFECT

Cost Elements	Program Costs: Design	Impl	T&I	Total
Systems Engineering	292	13	241	546
Programming	38	62	99	199
Configuration Control	68	19	151	238
Documentation	50	6	60	116
Program Management	28	6	30	64
Total	475	106	582	1162

Fig 10. Software Program Cost Ripple Effect (Ref 32)

--- PRICE SOFTWARE MODEL ---

DATE 04/24/78 TIME 11:20:54

SAMPLE CASE

MOBILE RADAR

SCHEDULE EFFECT SUMMARY

ACTIVITY LENGTH IN MONTHS

COMPLEXITY = 1.250	DESIGN	IMPL	T & I	TOTAL
SPECIFIED SCHEDULE	0.0	0.0	0.0	16.0
(OVERLAP)	(0.0)	(0.0)		
TYPICAL SCHEDULE	9.6	10.0	13.9	21.9
(OVERLAP)	(5.7)	(5.9)		

DEVELOPMENT COSTS

COMPLEXITY = 1.250	DESIGN	IMPL	T & I	TOTAL
SPECIFIED SCHEDULE	596.	146.	733.	1475.
TYPICAL SCHEDULE	500.	110.	611.	1222.
ESTIMATED PENALTY	96.	36.	122.	253.

Fig 11. PRICE-S Schedule Effects Summary Output (Ref 32)

values can vary from .5 to 11.0, with lower values describing the rather simplistic programming functions of mathematical operations, and larger values for the more complex requirements such as interactive (man/machine) interfaces. Table I provides a brief description of the various application types and appropriate APPL values. The actual proportion of instructions in each application category can be entered via the MIX input and the model will calculate an appropriate (weighted sum) value for APPL. APPL represents inherent project complexity independent of variations in other parameters. APPL values are cross-checked by the model with respect to the configuration of equipment provided.

Device types (TYPES) indicates the number of types of input/output equipments required for system operation. The quantities of input/output devices of the various types specified (DAT-data storage and retrieval, ONL-on-line communications, REA-real time command and control, INT-interactive operations) are entered as the Quantity (QTY) parameters. Inconsistencies during cross-checking will cause error notification to the user.

The cost escalation factor (ESC) can be used to reflect the expected economic inflation rate. The inflation factor is applied in accordance with the project schedule information. The model also contains an internal table of projected yearly inflation rates which can be selected.

The Function (FUNCT) and Level (LEVEL) variables are optional inputs which can be employed to estimate program size

TABLE I

PRICE-S Typical Application Values (Ref 32)

INSTRUCTION MIX

APPLICATION TYPE	WEIGHT	IDENTIFYING CHARACTERISTICS
OPERATING SYSTEMS	10.95	TASK MANAGEMENT. MEMORY MANAGEMENT. HEAVY HARDWARE INTERFACE. MANY INTERACTIONS. HIGH RELIABILITY AND STRICT TIMING REQUIREMENTS.
INTERACTIVE OPERATIONS	10.95	MAN/MACHINE INTERFACES. HUMAN ENGINEERING CONSIDERATIONS AND ERROR PROTECTION VERY IMPORTANT.
REAL TIME COMMAND AND CONTROL	8.46	MACHINE TO MACHINE COMMUNICATIONS UNDER TIGHT TIMING CONSTRAINTS. QUEUING NOT PRACTICABLE. HEAVY HARDWARE INTERFACE. STRICT PROTOCOL REQUIREMENTS.
ON-LINE COMMUNICATIONS	6.16	MACHINE TO MACHINE COMMUNICATIONS WITH QUEUING ALLOWED. TIMING RESTRICTIONS NOT AS RESTRICTIVE AS WITH REAL TIME COMMAND AND CONTROL.
DATA STORAGE AND RETRIEVAL	4.10	OPERATION OF DATA STORAGE DEVICES. DATA BASE MANAGEMENT. SECONDARY STORAGE HANDLING. DATA BLOCKING AND DEBLOCKING. HASHING TECHNIQUES. HARDWARE ORIENTED.
STRING MANIPULATION	2.31	ROUTINE APPLICATIONS WITH NO OVERRIDING CONSTRAINTS. NOT ORIENTED TOWARD MATHEMATICS. TYPIFIED BY LANGUAGE COMPILERS, SORTING, FORMATTING, BUFFER MANIPULATION, ETC.
MATHEMATICAL OPERATIONS	.86	ROUTINE MATHEMATICAL APPLICATIONS WITH NO OVERRIDING CONSTRAINTS.

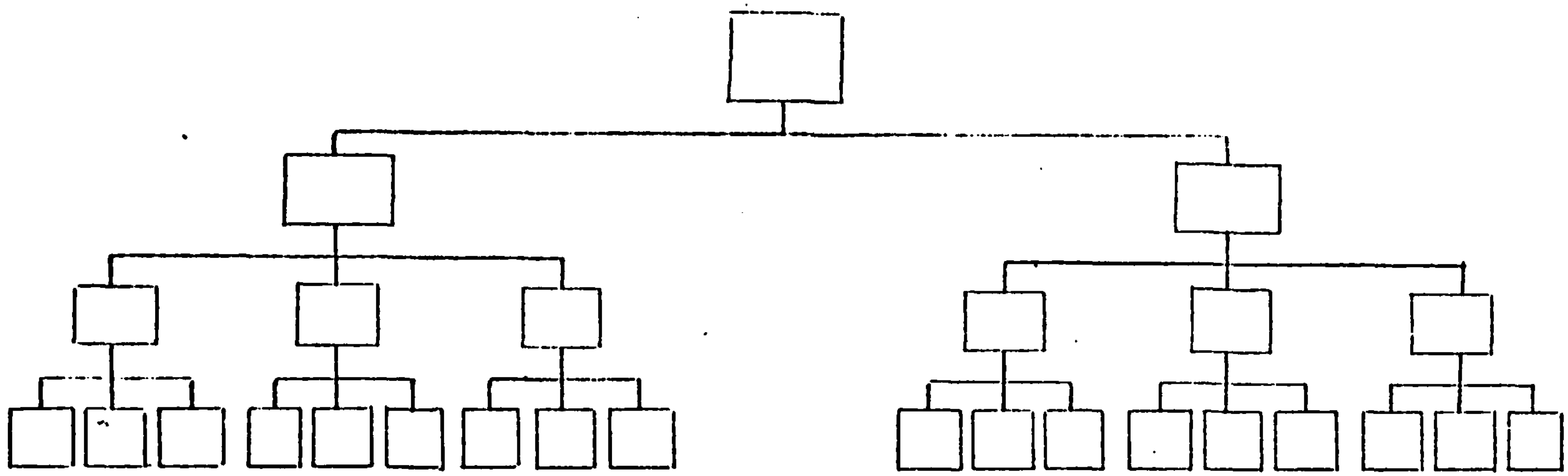
from program functional flow diagrams (Figure 12).

The Instruction (INST) parameter is the total number of executable machine-level instructions. PRICE S computes an estimate of the minimum number of instructions based on the system description provided for cross-check comparison to the INST input. In the Design-to-Cost mode, a system description and schedule along with a Target Cost (TARCST) are input and the total number of machine-level instructions is estimated. Typical values for converting High Order Language instruction quantities to equivalent machine-level instructions are provided in Table II.

An Integration variable (INTEG) is used to describe the extent of system-level integration required when separate subsystems are combined into a total operational system. INTEG is required only when a separate system-level integration activity is being modeled.

The Multiplier variable (MULT) can be used to multiply all cost factors by a specific multiple factor. The variable is used to include such items as profit, general and administrative changes, and Research and Development costs. MULT can also be used to convert the model outputs from dollars to man-months. Individual multipliers are also available for each cost element, thus providing the user with the capability to tailor the model to alternative cost reporting systems.

The eight elements of New Code (CODE) and New Design (DESIGN) are used to specify the proportion of instructions



TREE LEVEL	# OF FUNCT. MODULES	PRODUCT
0	1	0
1	2	2
2	6	12
3	18	54
	27	68

LEVEL = 68/27 = 2.51

STRUCTURE = 1.379

Fig 12. Software Functional Flow Diagram (Ref 32)

TABLE II

PRICE-S Typical HOL Instruction

Conversion Values (Ref 32)

Language	Conversion Ratio From Higher Order Language To Machine Level Language
COBOL	3 to 1
FORTRAN	5.5 to 1
JOVIAL	4 to 1
PL1	9 to 1
ATLAS	5 to 1, 12-15 to 1 due to different versions of ATLAS
MICRO-Code	3 to 1
ALGOL	10 to 1
UNIX-C	3 to 1
APL	15 to 1
PRIDE	5 to 1
FLOD	10 to 1
IFAM	13 to 1
CMS II	2.8 to 1
ASSEMBLY	1 to 1
COMPASS	.65 to 1

in each application class that requires new coding and new design effort respectively. The remaining proportions of CODE and DESIGN which are unspecified are assumed to already exist. The PRICE-S model will automatically estimate the amount of effort required to adapt existing design and code to the total software development.

Platform (PLTFM) is a variables which relates the system being developed to the specifications which must be met. PLTFM is a measure of the transportability, reliability, testing, and documentation required by contractual performance requirements. PLTFM relates the cost of software development to the requirements of the environment in which the software must operate. Table III contains a list of typical PLTFM values.

The Resource variable (RESO) is used to incorporate the effects of skill level, experience, productivity, efficiency, overhead, and labor rates for individual organizations on software development costs. This variable relates the scope of the work to the group doing the work. RESO values can range from 2.0 to 6.0, but normally center around a value of 3.5. RESO tends to remain essentially constant within a particular organization for a given class of projects. When known historical project costs are specified (TARCST), and the model is run in a reverse (ECIRP) mode, RESO values are calculated consistent with the project description provided.

Complexity (CPLX) is a schedule related variable which relates the relative difficulty of the programming task to the normal time required for its accomplishment. This

TABLE III

PRICE-S Typical PLATFORM Values

(Adapted from Ref 32)

Operating Environment	PLTFM
Production Center--Internally Developed Software	0.8
Production Center--Contracted Software	1.0
Military Mobile (Van or Shipboard)	1.4
Commercial Avionics	1.7
MIL-Spec Avionics	1.8
Unmanned Space	2.0
Manned Space	2.5
Military Ground	1.2

variable identifies complicating or simplifying factors which may be applicable to the project. The CPLX value starts with a base value of 1 and is adjusted up or down, based on the typical values provided in Table IV. If in normal operation CPLX and DSTART (date design effort starts) are specified, the model will generate a typical project schedule. If additional schedule data are input by the user, the model will compare the additional dates with the typical schedule and compute cost penalties associated with acceleration, stretch-out, or overlap deviations.

UTIL (Utilization) is a parameter which identifies the fraction of available hardware processor speed and memory capacity that is used. UTIL is a sensitive cost and schedule

TABLE IV

PRICE-S Typical COMPLEXITY Values

(Ref 32)

Typical Complexity Adjustments	
Personnel	ΔCPLX
Outstanding crew, among best in industry	-.2
Extensive experience, some top talent	-.1
Normal crew, experienced	0
Mixed experience, some new hires	+.1
Relatively inexperienced, many new hires	+.2
Product Familiarity	
Old hat, redo of previous work	-.2
Familiar type of project	-.1
Normal new project, normal line of business	0
New line of business	+.2
Complicating Factors	
New hardware	+.1
New language	+.1
More than one location/organization	+.2
Hardware developed in parallel	+.3
Many changing requirements	+.3
State-of-art advancement	+.4 to +.6

variable for values in excess of 0.75. This factor reflects the demand on software and programmers to adapt to the speed and memory constraints of hardware limitations.

The YEAR variable is used to establish a calendar reference for the model in initializing appropriate economic factors. By changing YEAR, the user can orient the model to any period of time desired.

The PRICE Software Model (PRICE-S) allows engineers, managers, and cost estimators to obtain assessments of manpower, schedule and budget requirements for computer software development. The model provides procedures to enable project description with a small set of cost drivers, and to permit calibration for individual organizations. The availability on commercial time-sharing systems provides for alternative evaluations with turn-around times of as little as five minutes. Although the PRICE-S model, which is depicted in Figure 13, has been designed so that it can be operated effectively with very little available information, it also provides considerable flexibility for more detailed studies which explore the effects of other related factors on software costs.

The first two chapters have provided a discussion of the Air Force computer resource acquisition and management policies and procedures. An overview of software cost-related factors and the RCA PRICE-S software cost estimation model was provided to establish a baseline for the analysis to follow. In the next two chapters an overview of the historical data bases will be provided with a detailed analysis of the research results.

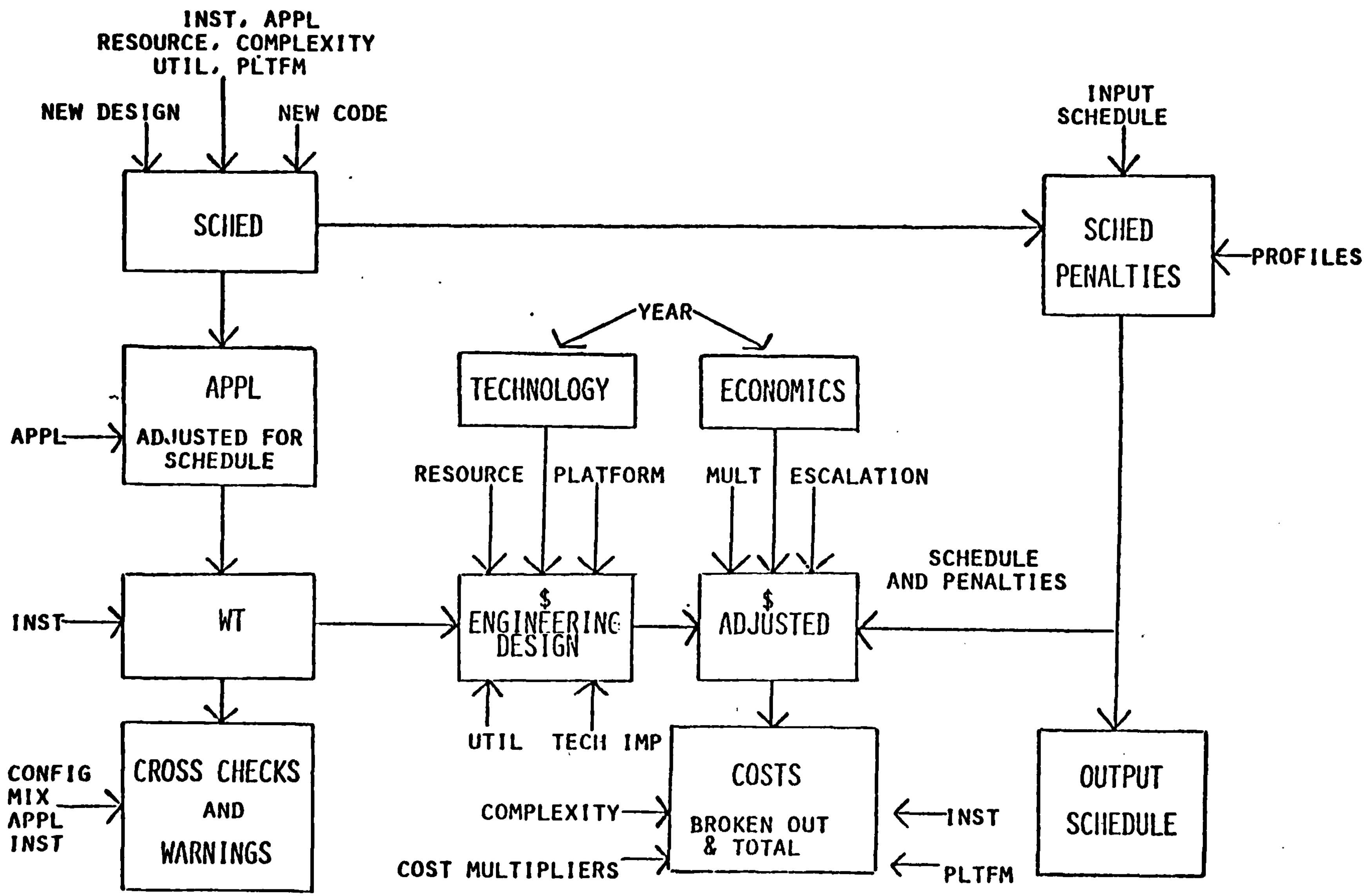


Fig 13. PRICE-S Cost Estimation Model Operation

III. Research Methodology

The purpose of this research is twofold. First, it is an effort to provide those involved in the management of software development projects and, in particular, those involved with software cost estimation and economic analysis with a basic understanding of the management process as it is presently practiced. Second, this research was conducted to determine the feasibility of implementing a common software cost estimation methodology, such as the RCA Price-S system, across all facets of Air Force software acquisition.

Data Collection

The data collection consisted of a literature search, combined with personal interviews relating to software development techniques and estimated and actual costs.

Collection Technique. Interviews were conducted with management and staff personnel at several Air Force commands and agencies to obtain applicable data, to discuss experiences in estimating development costs, and to identify other sources of data. Among the agencies interviewed were the following:

AF Aeronautical Systems Division, Wright-Patterson AFB, Ohio

AF Data Systems Design Center, Gunter AFS, Alabama

AF Electronic Systems Division, Hanscom AFB, Massachusetts

Air University, Maxwell AFB, Alabama

HQ USAF Logistics Command, Wright-Patterson AFB, Ohio

RCA Corporation, Cherry Hill, New Jersey

These interviews were accomplished through personal visits and by telephone. The interviews were conducted to obtain the quantitative data necessary to manipulate the model. From the commencement of the study effort, it was recognized that problems would be encountered with the collection of detailed historical data.

The time constraints and limited data availability for this study effort did not allow for the evaluation of a large number of historical projects or for extensive data analysis. Therefore, all conclusions, recommendations and observations contained in the study should be viewed keeping in mind the limited number of projects which have been studied. Since the limited number of systems analyzed do not constitute an adequate sample, no statistical analysis was applied to the results. Statistical indices could be misleading because of the limited sample size, therefore, any comparisons were made by using direct observations or percentage differences.

The accuracy and completeness of the input data varied widely from project to project. This accuracy and completeness must be considered when evaluating the results produced by the study effort.

Data Required. This study describes the effort on the part of the researcher to validate and calibrate the Price-S model for the gamut of Air Force software, including avionics, command and control and management data systems. The basic

problem was the identification of the "calibrated" values of the subjective model input parameters required to obtain an accurate estimate of project cost. It is hoped that this effort will provide assistance to future analysts, who may use the model.

The approach used by the researcher was to identify, through the interview process described, software development projects in each of the critical areas for which adequate historical cost information existed. Data relating to the input parameters for the PRICE-S model was collected for each of the identified systems. Utilizing the input data obtained for each of the software systems, outputs from the PRICE-S model were obtained for comparison with the historical cost data gathered during the interview process.

Collection Categories. Most management data system applications for the Air Force are implemented under the control of the Air Force Data Systems Design Center (AFDSDC). The primary language used is COBOL. The formalized procedures for the development of these systems, which are usually accomplished by Air Force personnel, were described in Chapter II--General Purpose Systems. Time and memory efficiency requirements for these systems are seldom severe and the systems are usually characterized by a high degree of I/O relative to computation. A majority of the management data systems are based on transaction-oriented processing to update files. Because of this relatively simplistic programming environment, productivity is normally very high for these systems. These

management data system applications on a cost-per-source line basis are usually less costly than non-business applications, since these applications are normally much less complicated than other applications such as command and control and avionics (Ref 21:80).

Most Air Force command and control systems are acquired under the direction of the Electronic Systems Division (ESD) of the Air Force Systems Command. The formalized procedures for the development of these systems (usually contractor developed) was also described in Chapter II--Embedded Systems. Most of this software is targeted for either large mainframes or ground-based mobile minicomputer systems. The standard language used in these applications is JOVIAL. There are a number of diverse functions of software involved in command and control applications such as data base management, information retrieval and display generation. Analysis of overall command and control applications indicate that productivity was less as compared to the other applications because of the usually larger size of the systems (Ref 21:82).

Avionics software is generally divided into three categories: on-board flight programs, simulation, and Automatic Test Equipment. Air Force avionics systems are generally acquired by the Avionics Systems Division (ASD) of the AF Systems Command in accordance with the embedded systems policies and procedures described in Chapter II. Operational flight programs usually have to operate in real-time, quick response, memory-constrained environments, which has the

impact of lowering development productivity. In addition, OFPs usually require a great amount of testing in as much as a number of them are life critical; this too leads to decreased productivity as compared with other types of applications. There is currently a low degree of HOL implementation for OFPs as compared with simulation and ATE, however, the trend is toward increased utilization of HOLs, which should increase productivity for this type application (Ref 21:83).

Data Base Description

Data was gathered on 18 Air Force software development projects in sufficient detail to identify values for the various PRICE-S input parameters. These 18 systems consisted of 6 management data systems (2 each from AFDSDC, HQUSAFLC, and Air University), 9 command and control systems (3 from ESD and 6 from a September 1979 AFIT thesis [Ref 12] done by Captain Cooper, and 3 avionics systems (from the ASD Avionics laboratory). Data ranged from very complete system descriptions for the avionics and command and control systems to less detailed, more general overall descriptions for the management data systems. Developmental time frames ranged from as early as 1972 up to and including the present.

System Descriptions. No attempt will be made by the researcher to identify specific systems by name. What is provided in the following descriptions is a brief overview of each of the systems studied. This effort is intended only

to give the reader some insight to the various applications so that future users of the PRICE-S system might obtain a feel for the model parameters as they relate to specific applications.

The first of the ESD command and control systems is a large ground-based radar system which will be operated by Air Force personnel to provide warning of a ballistic missile attack against the U.S. The second ESD system provides for maintenance and upgrade of an air defense system to provide for additional communications links, and improved communications, command, and weapons control. The third ESD system is a tactical mobile communications system which will provide automated assessment of communications channels and identification of corrective action in case of malfunctions.

The first of the Cooper Survey systems involved the acquisition of a large-scale command and control system including the software for operations, displays and control of a modularly integrated, world-wide operation. System number two involved upgrading the capabilities of a large-scale, long-range radar system. Third was a program to develop an interface capability between operations and intelligence systems to provide for real-time integration of intelligence data in support of air battle management functions. The fourth of these systems is an intelligence threat detection/classification system which involves real-time processing of surveillance data. System five involved the development of a communications processing system with satellite, radio and

ground capabilities, all operating under the control of a central computer facility. The final of the Cooper Survey systems was a program for development of an integrated command-wide digital record communications system for command and control support requirements. All of the command and control systems were developed by contractor personnel with guidance by typical Air Force management activities.

The management data systems include first a system which provides the capability to develop, tailor, and communicate operations plans. The system includes modules which provide transaction update and modification capabilities for manpower, logistics and operations. The second of the management data systems is an automated system for the collection, recording, and computations relating to management engineering data.

The third in the area of management data systems provides a means to monitor the modification status of selected commodities. The system is designed to provide managers with a monthly status in terms of units and manhours. Outputs are used to control accumulated backlogs, resolve shortages, and assure desired configuration improvements. The fourth in the line of management data applications is a system for the maintenance of records on government-furnished equipment. The system tracks quantities and consumption rates related to specific contracts. Item data related to reparable inventory, quantities-in-work, production, and shipments is maintained.

The fifth of the management data systems in an interactive wartime simulation model. The model is used for training

purposes and involves planning, selecting and employing forces in a wartime simulation environment. Monte Carlo success/failure determinations for specific events are used throughout the model. The final management data system is an interactive maintenance management simulation. The system is used to access skills in managing manpower and resources to obtain increased efficiency. All management data systems contained in the study were developed in-house by Air Force programming personnel.

Finally, the three avionics systems include two contractor developed and one in-house developed systems. The first of the avionics systems involved the conversion of an existing inertial navigation software system to an updated computer system. The conversion effort involved the alignment and navigational algorithm portions of the software. The second avionics system application was the development of an inertial navigation system simulator. The program involves the solution of differential equations, the initialization of matrix calculations, and the plotting of results. The final avionics system consisted of development of software for an airborne electronic radar system. Processing includes navigation update, antenna pointing, and data processing for mapping and terrain modes, and the control of associated input/output functions.

Model Application. Since the user can easily manipulate the model, erroneous estimates are a distinct possibility. Because of this, a procedure was adopted whereby specific

parameter values were logically and sequentially introduced. First, all model input files were built from the system descriptions provided in each of the four data bases: ASD-avionics, ESD-command and control, Cooper Survey-command and control, and AFDSDC/AFLC/AU-management data systems. All of the PRICE-S model input forms are contained in Appendix A of this report.

Files were built and checked with the Editor Function of the PRICE-S model. One software system from each of the four data bases was selected and run with nominal values of the input variables and an unconstrained schedule to obtain a general estimate of the project cost. These generalized estimates were compared with actual historical cost data to give the analyst a feel for what the model, in its nominal configuration, was projecting for project cost. The general estimate also gave the analyst an impression of what input parameter adjustments would be needed for calibration and validation purposes. Each of the four selected software systems was subsequently subjected to an ECIRP or calibration run of the model to determine the organizationally-oriented values for the RESO variable.

Subsequent model cost projection runs were made for each of the software systems in each of the four data bases. In each of the subsequent runs, the historical schedule data was input along with the appropriate values of the complexity variable (obtained from analysis of system descriptions) and resource (obtained from the calibration exercise). The

initial run for each system was accomplished using the model SHORT option, which significantly reduces the volume of printed output. Results of these initial runs were checked for reasonableness and compared to previous results to determine any areas of significant differences.

After the analyst was convinced that no significant disparities existed in the estimates obtained, the model was rerun for each of the software systems in the data bases choosing the Sensitivity and SCHEDULE model options. The additional information provided by this run was used by the analyst in assessing the variations in predicted costs due to changes in the input variables, as well as the effects of the imposed schedules. Analysis of the four data bases were conducted and are presented in Chapter IV. A complete set of model cost estimate outputs are provided in Appendix B of this report.

To evaluate the model sensitivity to general variations in the input parameters, an exercise was conducted to compare generalized variations of each of the major variables on ultimate project cost. The results of this analysis are presented in Appendix C.

The procedure just described was followed to insure the analyst that all appropriate input data was captured and that parameter values were known prior to conducting comparisons between runs or when analyzing predicted costs or investigating discrepancies.

The following chapter will provide a comprehensive

analysis of the study results. Each software development will be analyzed in detail on its own and in conjunction with results of similar development efforts.

IV. Analysis and Results

The Problem

The purpose of this research was to examine, through an analysis of historical data on previous Air Force software development projects, the adequacy of the RCA PRICE-S cost estimation model to predict accurately costs for both embedded (e.g. avionics, command and control) and general purpose (management data) systems. The examination was specifically limited to the area of software cost estimation, and no effort was made to include computer hardware costs in the analysis. While a complete economic analysis for an automated system would, of necessity, have to include both the software costs and hardware costs (which could be substantial in a multi-site/multi-weapon system), the major problem area in past efforts has been in accurately predicting software development cost and schedule.

The objective of the introductory chapter was to provide an understanding of the basic DOD/Air Force computer system acquisition processes. Chapter II provided a background in the management techniques used for Air Force embedded and general purpose software system developments. Factors effecting software cost and the PRICE-S model philosophy and operation have been described in preparation for the following analysis.

Data Analysis

Each of the 18 systems in the four data bases (ASD, ESD, Cooper Survey, AFDSDC) for which historical development information and cost data was collected will be analyzed. In each of the data bases the input descriptions collected through the data collection interviews will be presented and discussed. The PRICE-S model input forms for each of the systems were prepared from the data collected during the interviews and are provided in Appendix A.

Values of the quantitative parameters (instructions, platform, mix, schedule data, etc.) were taken directly from the information provided to the analyst. Values for the qualitatively based parameters (resource, complexity, etc.) were estimated based on discussions with personnel providing the data. Each of the model output results are provided in Appendix B.

Embedded Avionics. Three avionics systems software applications which were briefly described in Chapter III were selected for analysis. A summary of the major input parameters for the three systems is contained in Table V. The total number of machine level executable instructions in each program was obtained from project personnel. The application is a general interpretation of the type of coding (see Table I) contained in the system. Development schedule information (starting and ending dates) were provided by project personnel, as were the percent of central processor unit (CPU) capacity used by the software system. The environment is the basic

TABLE V

Embedded Avionics Systems Basic Description

Project	Instructions	Application	Design Start	Test End	CPU Utilization	Environment
ASA	18000	Real Time	0573	1275	.50	Avionics
ASB	2600	Mathematical	1177	0278	.30	Simulation
ASC	73750	Real Time	0574	0579	.85	Avionics

operating environment for the software and relates to the PRICE-S model Platform variable.

Project ASA was selected for the model calibration (ECIRP) analysis. Results of the embedded avionics model exercise are summarized in Table VI. Using the actual cost for Project ASA of $425k and a complexity (CPLX) of 1.0, with the schedule data as provided, the calibrated value of the resource (RESO) variable was 2.787. Complete detailed model outputs for each of the systems can be found in Appendix B. RESO values for the two remaining projects were adjusted to reflect differences in programmer productivity/ efficiency as assessed by project personnel. Project B personnel were less experienced while Project C personnel were slightly more experienced in software coding.

Project A was considered a normal new project (CPLX-1.0). Project B was input at CPLX-.9 because it was considered a redo of previous work (but with less experienced personnel), and Project C at .9 also because of the similar experience level of the programming personnel. For Projects A and B, which were done in-house, MULT 1.0 was selected. A 12 percent additional cost factor was included for Project C (MULT-1.2) as an adjustment to account for contractor fees. Projects A and C were designed for airborne applications (PLTFM-1.7) while Project B was designed for a fixed ground application (PLTFM-1.0).

The PRICE-S model estimated cost for Project ASA was $428,000, which was only 1 percent above the actual historical

TABLE VI

Embedded Avionics Systems Model Results

Project	Est Cost ($k)	Act Cost ($k)	% Diff	RESO	CPLX	Act Sched (mos)	Model Sched (mos)	Penalty ($k)
Part I: ECIRP Calibration								
ASA		425		2.787	1.0			
Part II: Validation								
ASA	428	425	+1	2.8	1.0	31	17.1	74
ASB	11	12	-9	2.9	.9	3	2.1	0
ASC	2102	2000	+5	2.7	.9	60	21	771

cost. The model predicted schedule for Project A was 17 months as opposed to the 31 months indicated by the historical schedule information. The cost penalty for the schedule stretchout was approximately $74,000. This schedule difference indicated a possible inefficiency in resource utilization and suggests that future similar projects could be developed in less time. Model computed values for Project ASB cost and schedule figures for Project B were $11,000 and 2.1 months. Actual cost and schedule figures for Project B were $12,000 and 3 months. The sensitivity analysis portion of the model output shows that at RESO-3.0 and CPLX-1.0 the projected cost and schedule were $12,000 and 3 months respectively. Model estimated costs for Project ASC was $2,102,000 with a 21-month predicted schedule. Actuals for Project C were $2,000,000 and 60 months. Sensitivity analysis shows a predicted cost of $2,003,000 at a RESO value of 2.6, which would indicate a somewhat more qualified programming group. Of major concern in the analysis of Project C was the disparity between actual and typical schedules (60 months vs 21 months). Project personnel indicated that the reason for the extended software development schedule was the requirement for a simultaneous hardware development.

PRICE-S calibrated values for the application (APPL) parameter ranged from a high of 6.5 for the inertial navigation system (Project A) to a low of 1.4 for the ground-based simulation system (Project B). This data appears to confirm the generally held belief that embedded software is inherently

a more complex programming task (indicated by higher APPL values). Based on this relatively limited amount of data, it appears that the PRICE-S model can be used to adequately predict the cost of embedded avionics within acceptable limits. Given an estimated program size, a relatively accurate cost estimate for embedded avionics software should be obtainable with parameter values as follows: RESO = 2.9, CPLX = 1.0, and APPL = 6.

Embedded Command and Control-ESD. The most detailed data collected during this research was that provided by ESD on three major command and control software development projects. The three systems selected for analysis were described in Chapter III. The three systems' parametric descriptions are provided in Table VII.

Project A consists of 7 individual programs which make up the complete system. MIX category data was provided for each of the programs which comprise the total system. Also provided were details on the percentages of design and code which were required to complete the system. Schedule data for each of the systems was provided by project personnel. Based on preliminary ECIRP calibration and an analysis of the details provided in the programming environment and technique descriptions, it was decided to use parameter values of RESO = 2.9 and CPLX = .9 for this system.

Project B was a relatively smaller system consisting of 5 major programs. Again, MIX category data, percentages of design and code, and schedule start and finish information was

TABLE VII

Embedded Command and Control Systems (ESD)
Basic Description

Project A			
Program		Object Instructions	% CPU Utilization
EAA	Operating System	25500	63
EAB	Tactical Applications	180500	63
EAC	Simulation	70200	63
EAD	Structured Programming Tools	100000	63
EAE	Data Reduction Tools	56000	63
EAF	Radar Control Software	54950	75
EAG	Radar Signal Processor	7240	75
Program		% Code	Application
EAA	Operating System	100	OPR
EAB	Tactical Application		
	Real Time Monitor	9	OPR
	Radar Manager	44	INT
	Mission Control	40	REA
	Communications	7	STR
EAC	Simulation		
	Real Time	30	INT
	Target Generator	70	STR
EAD	Structured Programming Tools		
	Data Storage	50	DAT
	Data Manipulation	50	STR
EAE	Data Reduction Tools	100	STR
EAF	Radar Control Software		
	Operating Module	18	OPR
	Task Module	82	ONL
EAG	Radar Signal Processor	100	REA

TABLE VII: Project A, continued

Program	% New Design	% New Code	Design Start	Test End
EAA	100	100	0476	1177
EAB	90	100	0476	1277
EAC	40	100	0876	1077
EAD	50	100	0476	0177
EAE	100	100	0876	0178
EAF	70	100	0476	0877
EAG	100	100	0776	0877

Program	Programming Environment
EAA	Normal New Project
EAB	Normal New Project, New Hires
EAC	Normal New Project, New Hires
EAD	Familiar Project, New Hires
EAE	Familiar Project, New Hires
EAF	Normal New Project, New Hires
EAG	Hardware/Software Parallel Development

Technique:

Combination of High Order Language and Assembly Language Programming

FINAL COST: $10,700,000

Project B

Program		Object Instructions	% CPU Utilization
EBA	Operational	40500	75
EBB	Utility	60000	50
EBC	Data Reduction	18000	75
EBD	Simulation	15000	75
EBE	Test	18000	75

TABLE VII: Project B, continued

Program	% Code	Application
EBA Operational	100	OPR
EBB Utility	100	ONL
EBC Data Reduction	100	STR
EBD Simulation	100	REA
EBE Test	100	REA

Program	% New Design	% New Code	Design Start	Test End
EBA	50	100	0173	0676
EBB	50	100	0173	0676
EBC	50	100	0173	0676
EBD	50	100	0173	0676
EBE	50	100	0173	0676

Program	Programming Environment
EBA	Normal New Project, Timing Constraint
EBB	Normal New Project
EBC	Normal New Project
EBD	Normal New Project
EBE	Normal New Project, Timing Constraint

Technique:
Programmed in High Order Language

FINAL COST: $5,000,000

Project C

Program	Object Instruction	% CPU Utilization
ECA Application	250000	50
ECB Support	30000	50

TABLE VII: Project C, continued

Program	% Code	Application
ECA Application	100	REA
ECB Support	100	ONL

Program	% New Design	% New Code	Design Start	Test End
ECA	100	100	0474	0277
ECB	100	100	0474	0277

Program	Programming Environment
ECA	Normal, Some New Hires, Changing Requirements
ECB	Normal, Some New Hires, Changing Requirements

Technique:
Programmed in High Order Language

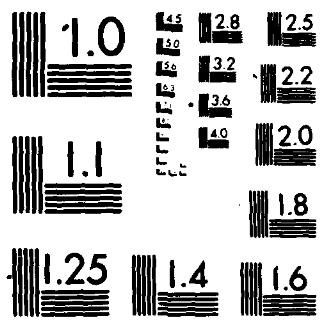
FINAL COST: $14,300,000

provided. Because the development involved some severe timing constraints and many changing requirements, the CPLX value was raised to 1.5 for this system. Also, project personnel indicated that many new hires were required which necessitated raising the RESO value to 3.2.

Project C was a rather large system which consisted of two major programs. Again, detailed size, design/code, MIX, and schedule data were extracted from system documentation. To accommodate the rapidly changing requirements activity which occurred during this system development, the CPLX value

AD-A083 713 AIR FORCE INST OF TECH WRIGHT-PATTERSON AFB OH SCHOO--ETC F/G 14/1
AN ANALYSIS OF THE RCA PRICE-S COST ESTIMATION MODEL AS IT RELA--ETC(U)
DEC 79 R E STEFFEY
UNCLASSIFIED AFIT/GSM/SM/79D-20 NL

2 of 3



MICROCOPY RESOLUTION TEST CHART

was increased to 1.3. The nominal RESO value of 2.9 was selected because the system was described as a typical contractor developmental effort.

The detailed model inputs can be seen in Appendix A. Appendix B contains the complete set of PRICE-S model outputs. Table VIII shows the consolidated results of the ESD model exercise. The combined estimate of project A was $10,934,000 with an actual cost of $10,700,000, showing a difference of +2%. Actual schedules versus model typical schedules for the 7 programs are relatively close. Project B shows a total estimated cost of $4,956,000 as compared to an actual cost of $5,000,000, or a difference of -1%. Actual schedule data may not be an accurate reflection of the program development. Single development start/test end dates were provided for each of the programs, indicating a simultaneous development effort. Considering the differences in size for the five programs, there were likely some differences in actual developmental time frames.

Project C actual cost was given as $14,300,000. The model estimated cost based on the input provided was $14,386,000 or a difference of +1%. The same argument regarding the actual schedule for Project B applies to Project C.

Based on the data provided, it appears that the PRICE-S model can accurately predict the cost of embedded command and control systems. Given an estimated program size, an adequate cost estimate should be obtainable with approximate parameter values of RESO = 3.0, CPLX = 1.0 and APPL = 8. It is

TABLE VIII

Embedded Command and Control Systems

Results - ESD

Project	Est Cost ($k)	RESO	CPLX	Act Sched (mos)	Model Sched (mos)	Penalty ($k)
EAA	825	2.9	.9	19	13.9	50
EAB	6665	2.9	.9	20	30.1	1663
EAC	799	2.9	.9	14	15.3	11
EAD	802	2.9	.9	9	12.0	121
EAE	465	2.9	.9	17	9.0	63
EAF	1106	2.9	.9	16	15.4	1
EAG	272	2.9	.9	13	11.2	4
TOTAL	10934	(+2%)				
ACTUAL	10700					
EBA	1934	3.2	1.5	41	29.5	149
EBB	1247	3.2	1.5	41	26.9	135
EBC	297	3.2	1.5	41	12.3	115
EBD	687	3.2	1.5	41	18.5	194
EBE	792	3.2	1.5	41	19.8	199
TOTAL	4956	(-1%)				
ACTUAL	5000					
ECA	13534	2.9	1.3	34	54.3	3665
ECB	852	2.9	1.3	34	22	93
TOTAL	14386	(+1%)				
ACTUAL	14300					

interesting to note the similarities in parameter values for the two types of embedded system software which would indicate that for gross estimating purposes, the same general values might be used in either instance. This line of reasoning will be pursued further in a later chapter.

Embedded Command and Control--Cooper Survey. The six embedded command and control software applications selected for analysis were described in Chapter III. A summary of the major input parameters for these six systems is contained in Table IX. This information was all extracted from the Cooper Survey instruments which were completed by project management personnel. Each of the systems was a real time command and control processing application. Since no data was available on actual program MIX composition, a value of APPL = 8.46 was assigned to each system to reflect the real time interactive application category. Platform (PLTFM) values of 1.2 military ground and 1.4 military mobile were assigned in accordance with the environmental descriptions. A 30 percent multiple factor was added to each system to account for contractor fees. Based upon the ECIRP calibration results from the ESD command and control systems baseline, RESO = 2.9 and CPLX = 1.0 values were assumed with adjustments made on the basis of the system descriptions.

Because of its extremely large size and extended development schedule, the Project CA resource and complexity values were adjusted upward to 3.0 and 1.1. Because of the relatively high values of CPU utilization for Projects CB and CE, the

TABLE IX

Embedded Command and Control Systems Basic Description

Cooper Survey

Project	Instructions	Application	Design Start*	Test End*	CPU* Utilization	Environment
CA	2750000	Real Time	0172	1279	.50	Military Ground
CB	20000	Real Time	0175	0676	.75	Military Ground
CC	250000	Real Time	0175	0677	.50	Military Ground
CD	140000	Real Time	0175	1276	.50	Military Mobile
CE	11000	Real Time	0175	1276	.75	Military Mobile
CF	156000	Real Time	0175	1277	.50	Military Ground

* Estimated from Project Descriptions
All Systems Considered to be Embedded Command and Control Systems

complexity factors for these two projects were raised to 1.3. Projects CD and CE appeared to have somewhat less experienced personnel performing the programming function, thus the RESO values were raised to 3.1 for these two systems.

Results of the Cooper Survey exercise are shown in Table X. Considering the limited amount of data available in the survey instrument, the estimated project costs are very close to the actuals, with two systems showing a 1% difference, three systems showing 3%, and one estimate which showed no difference. Discrepancies in the actual schedule versus "typical" schedule data were not as large for the embedded command and control systems as they were for the other data bases. It is interesting to note that whereas "typical" schedules were consistently shorter in the embedded avionics and management data systems, in the Cooper Survey systems, the typical schedules were longer than the actuals. This would probably indicate that additional resources are being added to these systems in an attempt to cut development time. The Cooper Survey results appear to confirm the ESD results which show that given an estimated program size, an accurate cost estimate for embedded command and control systems can be obtained with parameter values for RESO of 3.0, CPLX of 1.0, and APPL of 8.0.

Management Data Systems. Six management data systems software applications, which were briefly described in Chapter III, were selected for analysis. A summary of the major input parameters for these six systems is contained in Table XI.

TABLE X

Embedded Command and Control Systems Model Results

Cooper Survey

Project	Est Cost ($k)	Act Cost ($k)	% Diff	RESO*	CPLX*	Act Sched (mos)	Model Sched (mos)	Penalty
CA	82862	82500	+1	3.0	1.1	95	100.5	991
CB	991	1000	-1	3.0	1.3	17	19.8	42
CC	8220	8000	+3	2.9	1.2	29	41.3	1542
CD	5655	5500	+3	3.1	1.0	23	32.6	1053
CE	599	600	0	3.1	1.3	23	17.5	29
CF	6114	6300	-3	3.0	1.2	35	39.2	97

* Estimated from Project Description

TABLE XI

Management Data Systems Basic Description

Project	Instructions	Application	Design Start	Test End	CPU Utilization	Environment
DCA	315000	COBOL BUS	0278	0382	.10	AF Production Center
DCB	66000	COBOL BUS	0175	1276	.10	AF Production Center
ALA	27900	COBOL BUS	0578	0179	.10	AF Production Center
ALB	70140	COBOL BUS	1175	0878	.10	AF Production Center
AUA	30000	FORTRAN SIM	1077	0878	.10	AF Production Center
AUB	2900	FORTRAN SIM	0278	0778	.10	AF Production Center

The number of instructions for each program was obtained from systems documentation. The application is a broad based description of the type of programming contained in the system. Developmental schedule information was provided by project personnel or extracted from existing system documents. CPU utilization is generally insignificant because management data systems are normally run on very large general purpose processors in a multi-processing environment. The environment (PLATFORM variable) describes the basic software development methodology.

A slightly different approach was taken for calibrating the model for management data systems because of the lack of detailed MIX description data for these systems. Also, because this was the first effort at utilizing the model for Air Force management data systems, it was decided to ECIRP the entire data base. It was learned that there were no difficult or extenuating circumstances concerning the development of any of the six systems, thus a CPLX of 1.0 was assigned for each program. Since the detailed MIX category content information was not available, it was decided after discussion with Price Systems personnel that a 50%-50% mix of math operations and string manipulation could be used to approximate the management data systems application category. Results of the calibration can be seen in Table XII. RESO values ranged from a low of 1.3 to a high of 1.8, showing a somewhat consistent pattern which is markedly different from the embedded system values.

TABLE XII

Management Data System Calibration Results

Project	CPLX	APPL	RESO
DCA	1.0	1.588	1.476
DCB	1.0	1.588	1.287
ALA	1.0	1.588	1.352
ALB	1.0	1.588	1.688
AUA	1.0	1.588	1.454
AUB	1.0	1.588	1.783

CPLX - Set According to System Description
RESO - Calculated by PRICE-S

APPL - 50/50 mix Math Operations/String Manipulation

$\frac{9.04}{6} = 1.50$

Each of the systems (2-AFDSDC, 2-AFLC, 2-AU) was run through a model cost estimation exercise, the results of which are contained in Table XIII. Values of CPLX = 1.0, RESO = 1.5, and PLTFM = .8 were used for each of the systems. Actual costs of these in-house development efforts had to be determined indirectly. Developmental manhour figures were obtained from management personnel at each of the three development centers. Developmental manhours were converted to manyears based on the standard of 1728 productive manhours per manyear. The cost was then calculated based on the average 1978 Air Force programmer cost of $20,900 developed by the AFDSDC. The $20,900 figure was adjusted by a constant 6% inflation rate to the

TABLE XIII

Management Data Systems Results

Project	Est ($k)	Low	High	Development Hrs	Manyrs	Act ($k)	% Diff E	L	H	Sched (mos)	Sched (mos)	Penalty ($k)
DCA	416	376	460	33450	19.5	407	+2	-8	+13	49	11.4	181
DCB	76	69	82	6243	3.6	62	+22	+11	+32	23	6.3	25
ALA	36	33	39	2575	1.5	31	+16	+6	+25	8	4.4	3
ALB	102	93	111	12253	7.1	122	-19	-31	-9	33	6.4	45
AUA	38	35	42	3310	1.9	37	+2	-5	+13	10	4.6	5
AUB	7	6	7	760	.4	8	-14	-33	-14	5	1.9	1

1. CPLX Estimated Based on System Description From Management Personnel
2. Manyears Calculated Based on 1728 Productive Manhours per Year
3. Actual Cost Based on Average Air Force Programmer Cost $20900 per Year (1978) Adjusted to Base Year of System (Constant 6%)
4. RESO = 1.5 CPLX = 1.0 PLTFM = .8 used for all systems

base year of the development project. Detailed model outputs are contained in Appendix B.

The management data systems ranged in size from 315000 instructions to a low of 2900 instructions, and from 19.5 man-years of development effort (AFDSDC system A) to a low of .4 manyears (AU system B). Actual scheduled developmental time ranged from five months to four years. Again, for the management data systems there is a discrepancy between the actual and the model "typical" schedule data. This discrepancy again shows a consistent stretchout of schedule for defense systems. It is not apparent whether this stretchout is a result of inefficiencies in programming resource utilization or is the effect of the limited manpower resources available for defense systems as opposed to civilian industry.

Estimated costs for the two AFDSDC systems were $416,000 and $76,000, while the actual costs were calculated at $407,000 and $62,000 respectively. The estimates for these two systems represented differences of +2 percent and +22 percent. However, if the sensitivity analysis information is taken into consideration, the low value (indicating reduced complexity and resource values of .1) shows the estimate is reduced to $69,000 for the second system, or a difference of +11%. For the two AFLC systems, if sensitivity analysis is taken into consideration, the results show differences of estimated to actual of +6% and -9%. This would indicate that AFLC system A was probably less complex or more productive programming personnel were used as opposed to system B, which appears to

be either of higher complexity or less efficiency. The Air University systems appear to conform to tne nominally assigned values with differences of +2 and -14 percent. The relatively small size of the AU system B accounts for the larger percentage difference.

It is apparent from the results of this analysis that the management data system is an inherently less complex programming task than the embedded system. It appears that software costs can be significantly reduced by using inhouse development resources; however, consideration must be given to manpower limitations and the capabilities and expertise required for developing the more complex systems. Based on the analysis of this limited number of systems, it appears that the PRICE-S model can be used to adequately predict the cost of management data systems within acceptable limits. Given an estimated program size, a relatively accurate cost estimation for management data systems should be obtainable with parameter values as follows: RESO = 1.5, CPLX = 1.0, and APPL 2.0. Certainly much more accurate estimates can be obtained as the system progresses and becomes more well defined.

Table XIV shows a cost distribution summary be development phase and by cost element for each of the data bases. The composite figures show a distribution of 37-12-51 for the development phases, which closely approximates the widely accepted industry rule-of-thumb of 40-20-40.

In conclusion, based upon the results of this analysis, the PRICE-S software cost estimation model appears to have

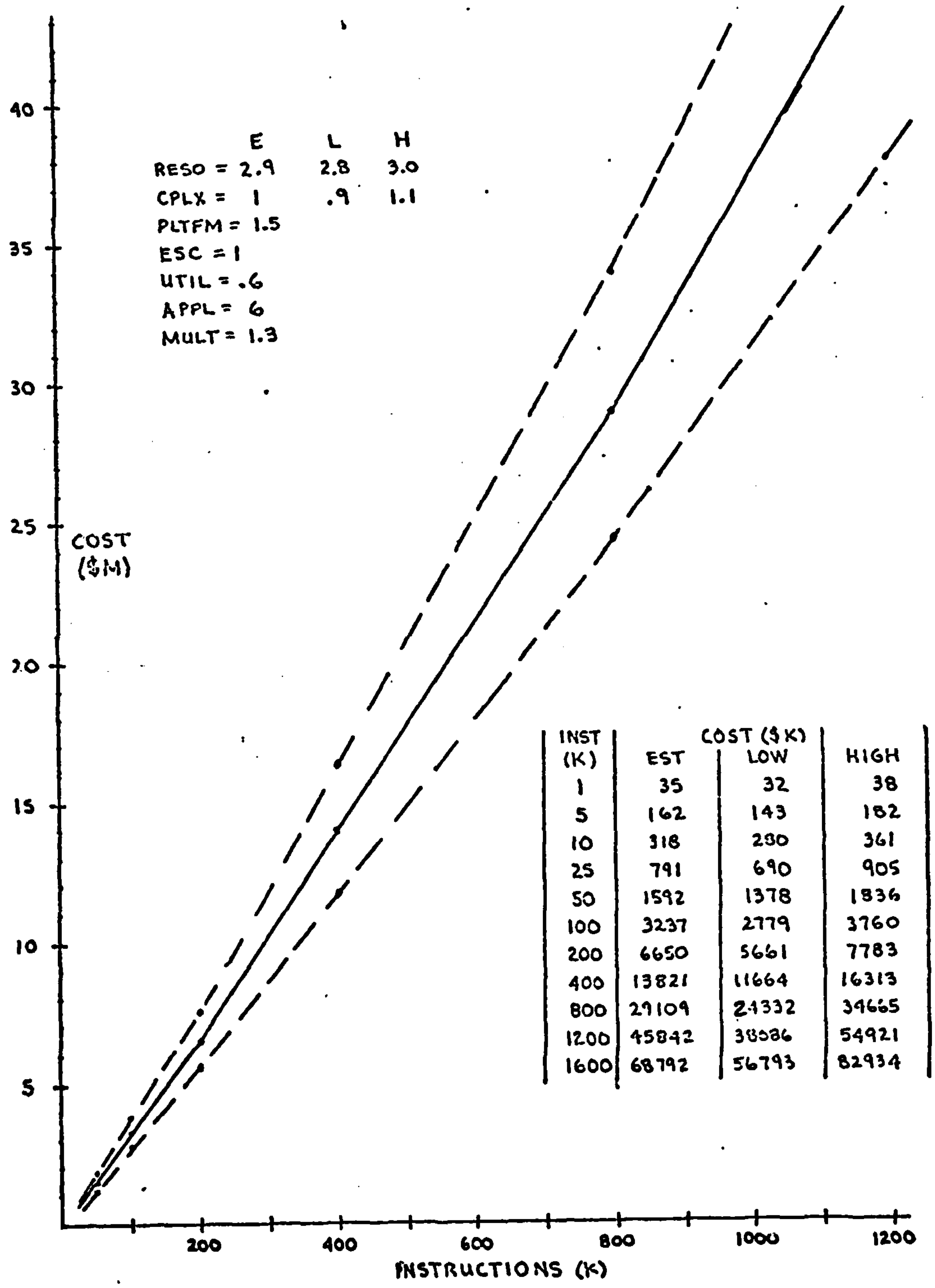
TABLE XIV

Cost Distribution Summary

Type Software	Percent Cost Element					Percent Life Cycle		
	SYSENG	PROG	CONFIG MGT	DOC	PMG MGT	DES	IMP	T&I
MGMT DATA SYS	50	22	16	6	6	39	12	49
CMD & CONT (ESD)	46	21	19	8	6	36	12	52
CMD & CONT (COP)	45	18	21	10	6	37	12	51
AVIONICS	45	20	19	10	6	38	11	51
COMPOSITE	46	20	19	9	6	37	12	51

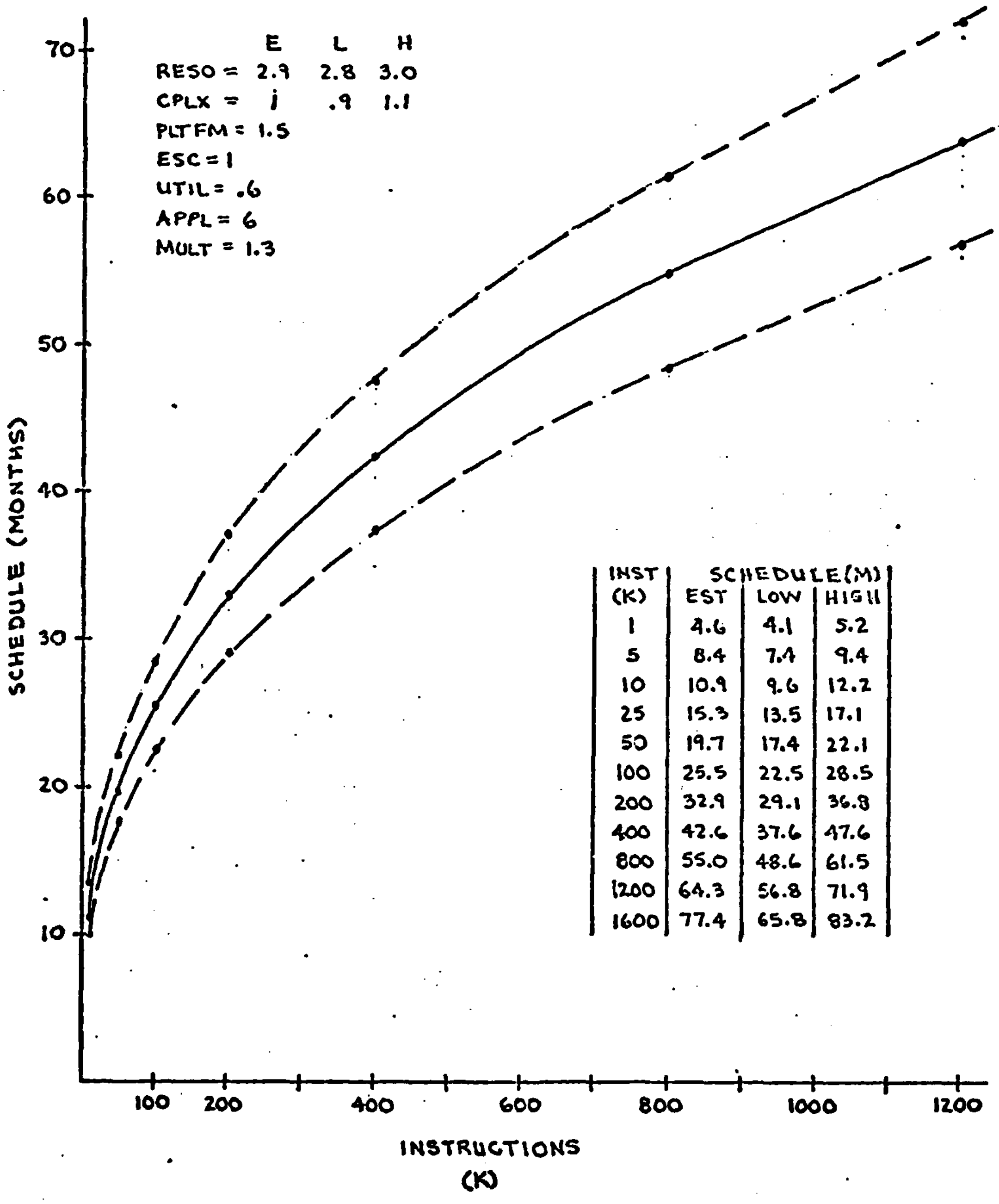
universal applicability across each of the major Air Force software functions. The flexibility of the model allows for accurate estimates of the simplest to the most complex programming tasks. The model can be manipulated easily to account for contracted as well as in-house development efforts. Allowable variations in the basic parameter enables the analyst to model an unlimited combination of resource utilization patterns. In the next section an illustration will be given of how to scope a rough approximation of program cost and schedule based on the results obtained from this preliminary examination of the PRICE-S model.

A Generalized Approximation Procedure. What is described here is an attempt at developing a gross software cost approximation based on the analysis results of this study. The analysis made a series of model runs for embedded and general purpose software developments ranging in size from 1,000 instructions to 1,600,000 instructions. In each case a generalized set of approximating variables was selected for both the embedded and general purpose projects. The estimated cost and schedule duration for each of the runs was recorded. Low and high values were obtained for each model run from the sensitivity analysis data in an effort to determine upper and lower bounds on the estimate cost and schedule. The effects of instruction quantities on cost and schedule duration for embedded software systems are shown in Figures 14 and 15. The general purpose systems cost and schedule duration relationships are shown in Figures 16 and 17.



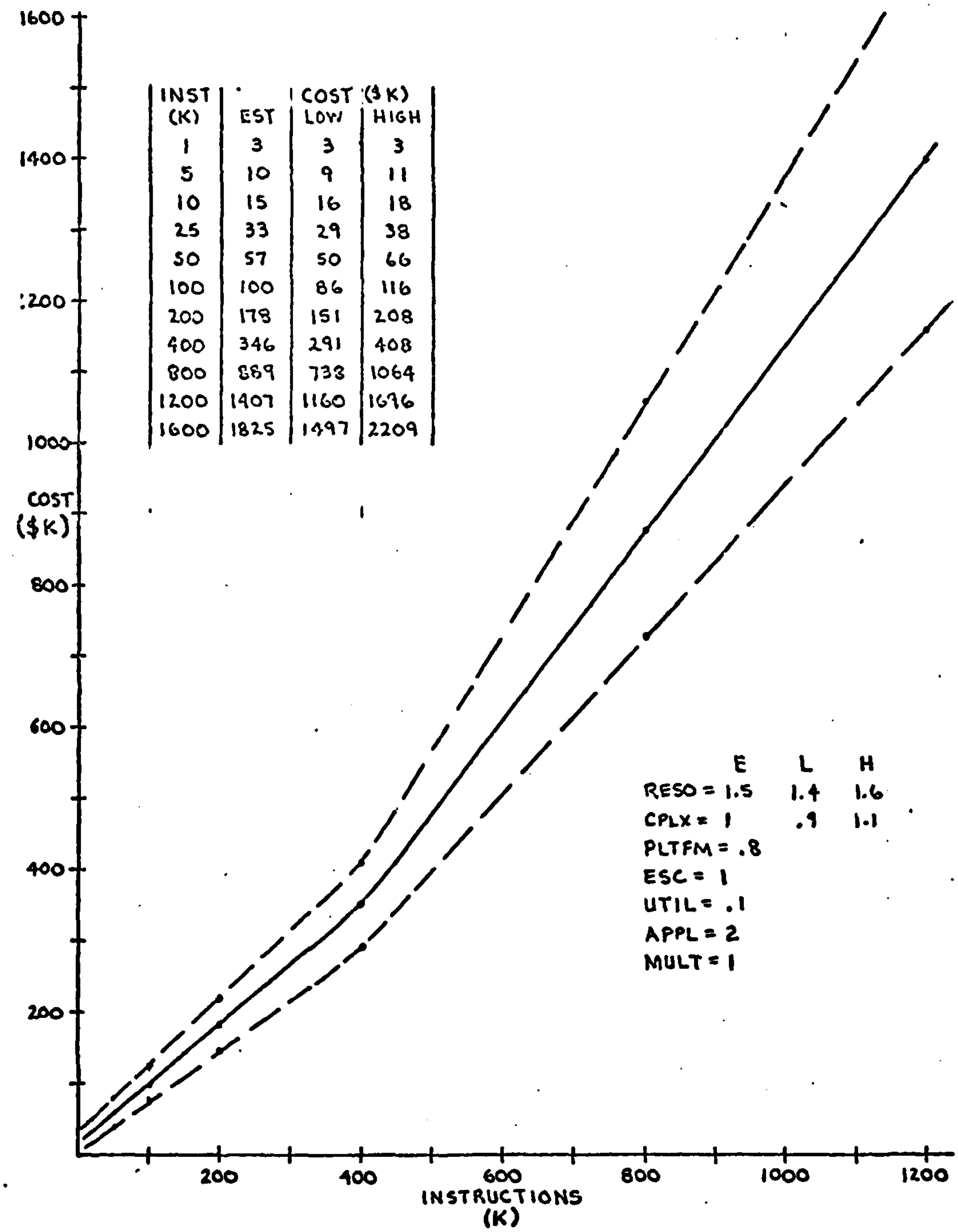
INST (K)	COST ($K) EST	LOW	HIGH
1	35	32	38
5	162	143	182
10	318	280	361
25	791	690	905
50	1592	1378	1836
100	3237	2779	3760
200	6650	5661	7783
400	13821	11664	16313
800	29109	24332	34665
1200	45842	38086	54921
1600	68792	56793	82934

Fig. 14. Effect of Instructions on Cost
for Embedded System Software



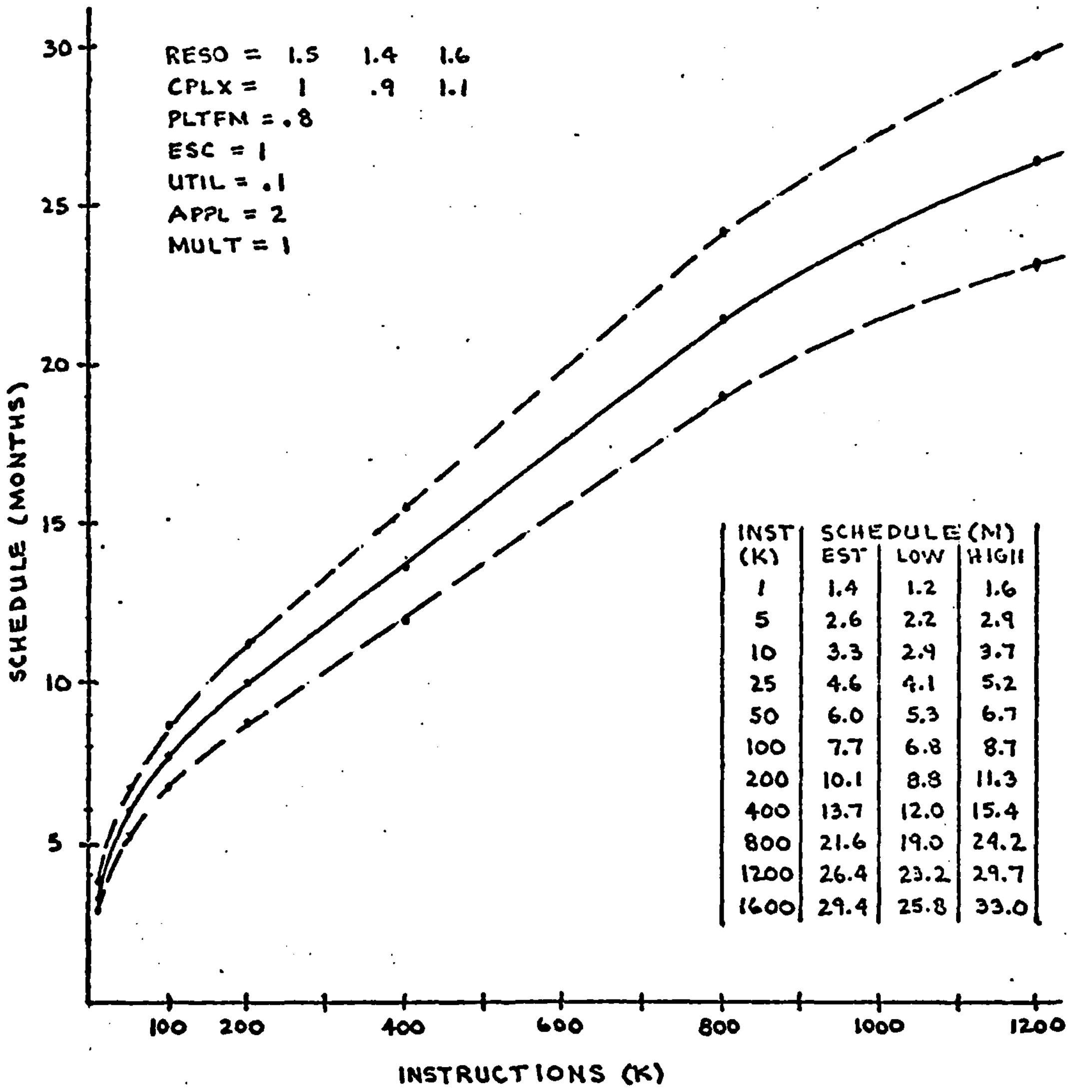
INST (K)	SCHEDULE (M) EST	LOW	HIGH
1	4.6	4.1	5.2
5	8.4	7.4	9.4
10	10.9	9.6	12.2
25	15.3	13.5	17.1
50	19.7	17.4	22.1
100	25.5	22.5	28.5
200	32.9	29.1	36.8
400	42.6	37.6	47.6
800	55.0	48.6	61.5
1200	64.3	56.8	71.9
1600	77.4	65.8	83.2

Fig 15. Effect of Instructions on Schedule for Embedded System Software



INST (K)	EST	COST ($K) LOW	HIGH
1	3	3	3
5	10	9	11
10	15	16	18
25	33	29	38
50	57	50	66
100	100	86	116
200	178	151	208
400	346	291	408
800	889	733	1064
1200	1407	1160	1696
1600	1825	1497	2209

Fig 16. Effect of Instructions on Cost for General Purpose Systems Software



INST (K)	SCHEDULE (M) EST	LOW	HIGH
1	1.4	1.2	1.6
5	2.6	2.2	2.9
10	3.3	2.9	3.7
25	4.6	4.1	5.2
50	6.0	5.3	6.7
100	7.7	6.8	8.7
200	10.1	8.8	11.3
400	13.7	12.0	15.4
800	21.6	19.0	24.2
1200	26.4	23.2	29.7
1600	29.4	25.8	33.0

Fig 17. Effect of Instructions on Schedule for General Purpose Systems Software

Given an estimate of program size, the plots can be used to obtain an approximate cost and schedule duration with a range based on variations in the RESO and CPLX variables. Figure 18 shows an analysis of a 400,000 executable instruction, contractor-developed JOVIAL embedded command and control program. Figures 14 and 15 show a development cost of $13,831,000 and a schedule of approximately 42.6 months. Using an industrial average programmer cost of $60,000 per year shows that the project involves 2764 manmonths of effort. This means that an average of 27.64 manmonths are required for each 1000 instructions in the JOVIAL language program. An average of 65 personnel per month are required to complete the project in the scheduled time. Based on the model resource utilization profiles, the maximum number of personnel available for any single month is 123. If a value of .2 is input at the value of the MULT variable, the model cost summary outputs will show equivalent manmonths instead of actual dollar costs.

Figure 19 shows an equivalent analysis for a 400,000 instruction in-house developed COBOL management data system. The charts reveal a 13.7 month schedule with a cost of $346,000. Using the $20,900 average programmer cost figure, the project requires approximately 198.6 manmonths of effort. This figure translates to about 1.5 manmonths per 1000 instructions of COBOL code. The average number of personnel required per month of the project is 14.5, with a maximum of 27.55 available in any single month. Setting the model MULT variable equal to .6 will result in a cost summary output in manmonths.

400,000* Executable Object Instructions

$\frac{400,000}{4}$ = 100,000 JOVIAL instructions (from conversion table)

42.6 Months Development Time (from Figure)

$13,821,000 Development Cost (from Figure)

1728 Manhours/Year

$60,000 /Manyear (Industry Average Programmer)

$5,000 /Manmonth

$\frac{13,821,000}{5000}$ = 2764 Manmonths

$\frac{10,000}{2764}$ = 36 Instructions/Manmonths

$\frac{2764}{100}$ = 27.64 Manmonths/1000 Instructions

$\frac{2764 \text{ manmonths}}{42.6 \text{ months}}$ = 65 Average number of people needed on project per month

1.9 Peak Average Resource (from Model Resource Distribution Profiles)

Maximum Number People Available for Any Single Month

65 x 1.9 = 123

Model Multiplier $\frac{1 \ (\$k)}{5 \ (\$k/\text{manmonth})}$ = .2 will give Cost Summary Output in Manmonths

* Estimate of executable instructions can be obtained by totaling all blocks in a system functional flow diagram and multiplying by 90 (model parameter for average number of instructions per block)

Fig 18. Embedded Software System Approximation (Contractor Development)

400,000* Executable Object Instructions

$\frac{400,000}{3}$ = 133,333 COBOL instructions (from conversion table)

13.7 Months Development Time (from Figure)

$346,000 Development Cost (from Figure)

1728 Manhours/Year (AF Standard

$20,900/Manyear (AFDSDC Standard Programmer)

$1742/Manmonth

$\frac{346,000}{1742}$ = 198.6 Manmonths

$\frac{133,333}{198.6}$ = 671 Instructions/Manmonth

$\frac{198.6}{133.3}$ = 1.5 Manmonths/1000 Instructions

$\frac{198.6 \text{ Manmonths}}{13.7 \text{ Months}}$ = 14.5 Average Number of People Needed on Project per Month

1.9 Peak Average Resource (from Model Resource Distribution Profiles)

Maximum Number People Available for any Single Month

$$14.5 \times 1.9 = 27.55$$

Model Multiplier $\frac{1(\$k)}{1.7\ (\$k/\text{manmonth})}$ = .6 will give Cost Summary Outputs in Manmonths

* Estimate of Executable instructions can be obtained by totaling all blocks in a system functional flow and multiplying by 90 (model parameter for average number of instructions per block)

Fig 19. General Purpose Software System Approximation (In-House Development)

The accuracy of this kind of methodology cannot be as precise as that gained from a detailed system analysis using the complete model procedure; however, this effort is only intended as an approximation technique designed to give the analyst a baseline figure from which to proceed.

Total Life Cycle Costing Effects. The present version of the PRICE-S model includes only software development costs. A complete economic analysis to include all life cycle costs might be required for various management or budgetary purposes. RCA PRICE Systems is presently involved in expansion of the existing model to include the front end requirements analysis, and back end maintenance and modifications portions of the complete software life cycle.

One recent study shows the total life cycle cost breakdown by phase in Figure 20 (Ref 28:18). The figure shows that the design phase makes up approximately 55 percent of the life cycle costs with maintenance and modification accounting for an additional 45 percent. The figure shows that the front end requirements analysis phase adds approximately 20 percent to the total life cycle cost. Using these gross approximations, Table XV was constructed showing the total life cycle cost estimates for the 18 systems utilized in this research effort. It can be seen that in any economic analysis where system acquisition or tradeoff requirements decisions might rest on total cost, the added requirements analysis and maintenance/modifications may prove to be important considerations in the final decisions.

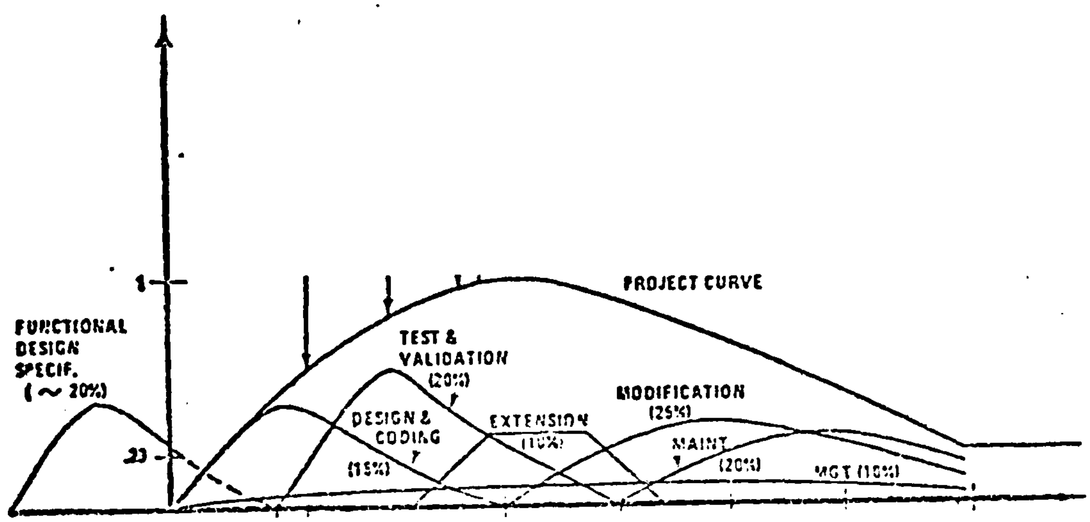


Fig 20. Total Life Cycle Cost Breakdown

This concludes the data analysis portion of this research effort. What was attempted, in the time available, was a PRICE-S model exercise looking at the three major Air Force software development areas of embedded avionics, embedded command and control, and management data systems. Utilizing historical cost, schedule and development data gathered on 18 Air Force software projects, model estimates were obtained and compared to the actual historical figures. Conclusions were drawn regarding the three application areas based on the results of the model output. Finally, an attempt was made to extend the results of the model exercise into a general estimation technique which included a look at total life cycle cost aspects of software development. In the next chapter the results of this research will be summarized and recommendations regarding follow-on activities will be discussed.

TABLE XV

Life Cycle Cost Estimation Summary

Project	RQMTS ANAL($k)	DEVEL($k)	MAINT/MOD($k)	TOT LCC($k)
EA	3976	10934	8946	23856
EB	1802	4956	4054	10812
EC	5231	14386	11770	31387
ASA	155	428	350	933
ASB	4	11	9	24
ASC	764	2102	1719	4585
DCA	151	416	340	907
DCB	27	76	62	165
ALA	13	36	29	78
ALB	37	102	83	222
AUA	13	38	31	82
AUB	2	7	5	14
CA	30131	82862	67796	180789
CB	360	991	810	2161
CC	2989	8220	6725	17934
CD	2056	5655	4626	12337
CE	217	599	490	1306
CF	2223	6114	5002	13339

(Model) Development Cost Approx 55% of TOT LCC
Maintenance and Modification Approx 45% of TOT LCC
Requirements Analysis Approx 20% of Development + Maintenance/Modification

V. Conclusions and Recommendations

Conclusions

Before managers can hope to control software costs, they must first understand how software costs are generated. Until the relationships between system parameters and software costs are recognized, control of software costs will be unobtainable. It is hoped that this research effort will provide some additional insight into the computer software acquisition, management and cost estimation processes.

The major objective of this research was an attempt to investigate and validate the applicability of the RCA PRICE-S software cost estimation model for possible use in conjunction with Air Force software acquisitions. A major portion of this report was, of necessity, devoted to providing a basic understanding of the Air Force computer software acquisition and management processes. It is the belief of the researcher that an accurate software cost estimation is not possible without a background in the processes by which costs are drawn.

Historical cost, schedule, and development data was collected from the three major Air Force software application areas of embedded avionics, embedded command and control, and management data systems. Discussions were conducted with project management personnel to familiarize the researcher

with current estimation practices. It was enlightening to discover the many and varied cost estimation techniques being employed not only across the different application areas, but also within the various areas. It is the belief of the researcher that the existance of too many estimation procedures is a contributing factor in the lack of progress toward more accurate cost estimation capabilities. Unless some positive action is taken to limit the number of procedures used to a few of the more promising, visible progress in refining the estimation process is unlikely.

The RCA PRICE-S system is a promising step in the direction of more accurate software cost and schedule predictions. Based on the results of this limited investigation, the PRICE-S system appears to have universal applicability to all phases of Air Force software acquisition. Results of the model exercise were encouraging. The predictions were highly accurate and well within acceptable limits. Model flexibility allows the analyst to develop estimates with very little system descriptive information. Model sophistication provides the capability to adapt the model to individual organizational operation techniques. The scope of the model input parameters allows a progressively more detailed and comprehensive cost estimate as the program advances through the development process. Data required to operate the model is available and can be obtained with a minimum of effort.

Recommendations

There are a number of factors regarding the model which require some additional investigation. The cost multiplier

globals (ATABLE) allow the user to modify the allocation of costs to each of the five cost elements in each of the three development phases. Further analysis needs to be conducted to ensure that the nominal model values accurately portray the distributions actually occurring in the embedded and general purpose areas for Air Force developments.

The resource allocation profiles which are computed based on the curve control globals (CTABLE) use three β distributions of the development costs through each of the three development phases. The combination of these three distributions may not be representative of the situation actually occurring in Air Force developments. Further analysis is needed to ensure that the nominal values are accurate or to develop more accurate portrayels of Air Force systems.

The PRICE-S model calculates what it refers to as a "typical schedule" with normal overlaps for the three activity phases (design, implementation, test and integration). This schedule is computed based on the size, type, and difficulty of the project described. Based on the input schedule identified by the analyst, a comparison is made with the internal model typical schedule and costs are adjusted to account for apparent accelerations, stretch-outs, and phase transition inefficiencies. Project ASC, for example, showed an actual schedule of 60 months and a typical schedule of 21 months. The result was a penalty cost of $771,000 for schedule stretch-out. The developer indicated that 60 months was required because the system involved simultaneous hardware and software

development. While this situation could be addressed through the complexity variable, it appears that the "typical schedule"/"penalty" effects on the cost estimate should be examined in more detail.

In order to accomplish the management data systems model exercise, it was necessary to assume an application (APPL) value based on discussions with project management and PRICE Systems personnel. Additional analysis is needed in this area to determine more accurately the management data systems mix category composition.

Finally, and most importantly, the software cost estimation process could be improved by the adoption of a systematic approach and the development of an integrated estimation methodology. Such a methodology should include:

1) Training and utilization of qualified personnel with experience and knowledge of the software acquisition, management, and development processes. The system must draw upon information accumulated from both the military and industry.
2) Evolving a systematic cost estimating procedure which includes a series of steps to:
   a) define the cost estimating task
   b) identify the resources required in the estimation process
   c) identify the estimation technique to be employed
3) Deriving software sizing techniques based on technical evaluation of the functional performance

requirements of the software.

4) Collecting and analyzing software cost data from new and existing systems, based on common definitions of data parameters to provide an historical cost element data base for derivation of cost estimates.
5) Employing a validated cost model supported by analysis of the historical cost data base.
6) Ensuring that the procedures for software cost estimating are rigorously and methodically followed.

BIBLIOGRAPHY

Bibliography

1. Aeronautical Systems Division. Management Guide to Avionics Software Acquisition. Technical Report, Vol III, Wright-Patterson AFB, OH. June 1976.

2. ________. Proceedings of the Avionics Software Workshop. Wright-Patterson AFB, OH. October 1975.

3. AFR 300-2. Management of the USAF Automatic Data Processing Program. Washington: Government Printing Office, August 1977.

4. AFR 300-12. Procedures for Managing Automatic Data Processing Systems, Documentation, Development, Acquisition and Implementation. Washington: Government Printing Office, September 1977.

5. AFR 300-15. Air Force ADP Software Project Manual. Draft. Washington: Government Printing Office, June 1975.

6. AFR 800-14. Acquisition and Support Procedures for Computer Resources in Systems. Washington: Government Printing Office, September 1975.

7. AF Select Committee on Computer Technology Potential. Study: Air Force Organizational Ability to Exploit and Manage Computer Technology. Washington: HQ USAF, July 1970.

8. Air Force Logistics Command. Project Pacer Flash. Wright-Patterson AFB, OH. September 1974.

9. Baker, W.F. Software Data Collection and Analysis: A Real Time System Project History. RAOS-TR-77-192. Gaithersburg, MD: IBM Corporation, June 1977.

10. Clapp, J.A. A Review of Software Cost Estimation Methods. Technical Report ESD-TR-76-271. Bedford, MA: Mitre Corporation, August 1976.

11. Commander's Digest. Volume 13, No. 33. Washington: Department of Defense, June 1973.

12. Cooper, V. "An Investigation of the Software Requirements Allocation Process in the Acquisition and Management of a Major Defense System." Unpublished master's thesis. Air Force Institute of Technology, Wright-Patterson AFB, OH. September 1979.

13. Deputy Assistant Secretary of Defense, Production Engineering and Material Acquisition. Tactical Computer Software Acquisition and Maintenance Study. Washington: Government Printing Office, October 1973.

14. Devenny, T.J. "An Exploratory Study of Software Cost Estimating at the Electronic Systems Division." Unpublished master's thesis. Air Force Institute of Technology, Wright-Patterson AFB, OH. July 1976. AD AO30162.

15. Devivo, R. Definition, Collection, and Organization of Software Parametric Data. Bedford, MA: Mitre Corporation, June 1977.

16. DOD Directive 5000.1. Acquisition of Major Defense Systems. Washington: Government Printing Office, December 1975.

17. DOD Directive 5100.40. Responsibility for the Administration of the DOD Automatic Data Processing Program. ASD(C). Washington: Government Printing Office, August 1975.

18. Fisher, David A. Automatic Data Processing Costs in the Defense Department. Washington: Institute for Defense Analysis, October 1974.

19. Gansler, Jacques S. Deputy Assistance Secretary of Defense, Material Acquisition. Remarks made before the Polytechnic Institute of New York Symposium on Computer Software Engineering. 20 April 1976.

20. Graver, C.A. Cost Reporting Elements and Activity Cost Tradeoffs for Defense System Software: Volume 1, Study Results. Santa Barbara, CA: General Research Corporation, November 1976.

21. Herd, J.H., J.N. Postak, W.E. Russell and K.R. Stewart. Software Cost Estimation Study. RADC-TR-77-220. Griffis AFB, NY: Rome Air Development Center, June 1977.

22. HQ Air Force Logistics Command. "Management of Weapon Systems Computer Resources." Briefing. Wright-Patterson AFB, OH. 1 May 1978.

23. Institute for Defense Analysis, Science and Technology Division. Electronics X: A Study of Military Electronics with Particular Reference to Cost and Reliability. Vol. 2. Washington D.C., January 1974.

24. Johns Hopkins University Applied Physics Laboratory. DOD Weapon System Software Management Study. Draft. Baltimore, MD: Johns Hopkins University Press, May 1975.

25. Mish, Russell and Marsha Fifner. Software Acquisition Acquisition Management Guidebook: Cost Estimation and Measurement. ESD TR-78-140. Santa Monica, CA: Systems Development Corporation, March 1978.

26. Mitre Corporation. DOD Weapon System Acquisition and and Management Study. Technical Report No. 6908, Vol. I. Bedford, MA: Mitre Corporation, May 1975.

27. Nelson, E.A. Management Handbook for the Estimation of Computer Programming Costs. Santa Monica, CA: Systems Development Corporation, 1965.

28. Putnam, J.H. "General Empirical Solution to the Macro Software Sizing and Estimating Problem," IEEE Transactions on Software Engineering, July 1978.

29. Schneider, J. "A Preliminary Calibration of the RCA PRICE-S Software Cost Estimation Model." Unpublished master's thesis. Air Force Institute of Technology, Wright-Patterson AFB, OH. July 1976. AD A046808.

30. The Air Force Budget. Washington: HQ USAF (AF/AC), February 1975.

31. TRW Software Series. Software Development and Configuration Management Manual. TRW-SS-73-07. Redondo Beach, CA: TRW Corporation, December 1973.

32. Unknown. Reference Manual: Price Software Model. Cherry Hill, NJ: RCA Price Systems, December 1975.

33. United States Air Force Project Rand. Air Force Command and Control Information Processing in the 1980s: Trends in Software Technology. R-1012-PR. Santa Monica, CA: Rand Corporation, June 1974.

34. Walker, W.H. "An Approach to Software Life Cycle Cost Modeling." Unpublished master's thesis. Air Force Institute of Technology, Wright-Patterson AFB, OH. December 1978. AD A064223.

35. Willmorth, N.W. Software Development Data Collection Survey of Project Managers. TM-5542/005/00. Santa Monica, CA: Systems Development Corporation, February 1976.

36. Wolverton, R.W. "The Cost of Developing Large Scale Software," IEEE Transactions on Computers, June 1974.

37. Zempolich, Bernard A. An Analysis of Computer Software Management Requirements for Operationally Deployable Systems. Washington: Industrial College of the Armed Forces Program, April 1975.

APPENDIX A

PRICE-S Model Input Forms

PRICE Software Model Input Worksheet

Filename: STEAA

Page 1 of 7

Title: PROJECT EAA

Application: EARLY WARNING RADAR OPERATION

Date: 4 OCT 79

								Optional	
Descriptors	INST	APPL	RESO	FUNCT	STRU	LEVEL		INTEG	
	25500	0	2.9						
Mix	MDAT	MONL	MREA	MINT	MMAT	MSTR	MOPR	MAPP	APPLS
	0	0	0	0	0	0	1		
New Design	DDAT	DONL	DREA	DINT	DMAT	DSTR	DOPR	DAPP	
	.001	.001	.001	.001	.001	.001	1		
New Code	CDAT	CONL	CREA	CINT	CMAT	CSTR	COPR	CAPP	
	.001	.001	.001	.001	.001	.001	1		
Device Types	TDAT	TONL	TREA	TINT					
	0								
Quantity	QDAT	QONL	QREA	QINT					
	0								
Schedule Data	CPLX	DSTART	DEND	ISTART	IEND	TSTART	TEND		
	.9	0476	0	0	0	0	1177		
Supplementary Information	YEAR	ESC	TECIMP	MULT	PLTFM	UTIL		TARCST	
	1976	1	1	1.3	1.2	.63			

Notes:

GC 1610 8/77

RCA

PRICE Software Model Input Worksheet.

Filename: STEAB

Page 2 of 7

Title: PROJECT EAB

Application: EARLY WARNING RADAR APPLICATION

Date: 2 OCT 79

								Optional	
Descriptors	INST 180500	APPL 0	RESO 2.9	FUNCT	STRU	LEVEL		INTEG	
Mix	MDAT 0	MONL 0	MREA .40	MINT .44	MMAT 0	MSTR .07	MOPR .09	MAPP	APPL8
New Design	DDAT .001	DONL .001	DREA .90	DINT .90	DMAT .001	DSTR .90	DOPR .90	DAPP	
New Code	CDAT .001	CONL .001	CREA 1	CINT 1	CMAT .001	CSTR 1	COPR 1	CAPP	
Device Types	TDAT 0	TONL	TREA	TINT					
Quantity	QDAT 0	QONL	QREA	QINT					
Schedule Data	CPLX 1.0	DSTART 0476	DEND 0	ISTART 0	IEND 0	TSTART 0	TEND 1277		
Supplementary Information	YEAR 1976	ESC 1	TECIMP 1	MULT 1.3	PLTFM 1.2	UTIL .63		TARCST	

Notes:

GC 1610 8/77

RCA

PRICE Software Model Input Worksheet

Filename: STEAC

Page 3 of 7

Title: PROJECT EAC

Application: EARLY WARNING RADAR SIMULATION

Date: 2 OCT 79

								Optional	
Descriptors	INST	APPL	RESO	FUNCT	STRU	LEVEL		INTEG	
	70200	0	2.9						
Mix	MDAT	MONL	MREA	MINT	MMAT	MSTR	MOPR	MAPP	APPL8
	0	0	0	.30	0	.70	0		
New Design	DDAT	DONL	DREA	DINT	DMAT	DSTR	DOPR	DAPP	
	.001	.001	.001	.40	.001	.40	.001		
New Code	CDAT	CONL	CREA	CINT	CMAT	CSTR	COPR	CAPP	
	.001	.001	.001	1	.001	1	.001		
Device Types	TDAT	TONL	TREA	TINT					
	0								
Quantity	QDAT	QONL	QREA	QINT					
	0								
Schedule Data	CPLX	DSTART	DEND	ISTART	IEND	TSTART	TEND		
	1.0	0876	0	0	0	0	1077		
Supplementary Information	YEAR	ESC	TECIMP	MULT	PLTFM	UTIL		TARCST	
	1976	1	1	1.3	1.2	.63			

Notes:

GC 1610 8/77

RCA

PRICE Software Model Input Worksheet

Filename: STEAD

Title: PROJECT EAD

Application: EARLY WARNING RADAR PROG TOOLS

Date: 2 OCT 79

								Optional	
Descriptors	INST	APPL	RESO	FUNCT	STRU	LEVEL		INTEG	
	100000	0	2.9						
Mix	MDAT	MONL	MREA	MINT	MMAT	MSTR	MOPR	MAPP	APPL8
	.50	0	0	0	0	.50	0		
New Design	DDAT	DONL	DREA	DINT	DMAT	DSTR	DOPR	DAPP	
	.50	.001	.001	.001	.001	.50	.001		
New Code	CDAT	CONL	CREA	CINT	CMAT	CSTR	COPR	CAPP	
	1	.001	.001	.001	.001	1	.001		
Device Types	TDAT	TONL	TREA	TINT					
	0								
Quantity	QDAT	QONL	QREA	QINT					
	0								
Schedule Data	CPLX	DSTART	DEND	ISTART	IEND	TSTART	TEND		
	.8	0476	0	0	0	0	0177		
Supplementary Information	YEAR	ESC	TECIMP	MULT	PLTFM	UTIL		TARCST	
	1976	1	1	1.3	1.2	.63			

Notes: -.2 HOL

GC 1610 8/77

RCA

PRICE Software Model Input Worksheet

Filename: STEAE

Title: PROJECT EAE

Application: EARLY WARNING RADAR REDUCTION

Date: 2 OCT 79

Section								Optional	
Descriptors	INST	APPL	RESO	FUNCT	STRU	LEVEL		INTEG	
	56500	0	2.9						
Mix	MDAT	MONL	MREA	MINT	MMAT	MSTR	MOPR	MAPP	APPL8
	0	0	0	0	0	1	0		
New Design	DDAT	DONL	DREA	DINT	DMAT	DSTR	DOPR	DAPP	
	.001	.001	.001	.001	.001	1	.001		
New Code	CDAT	CONL	CREA	CINT	CMAT	CSTR	COPR	CAPP	
	.001	.001	.001	.001	.001	1	.001		
Device Types	TDAT	TONL	TREA	TINT					
	0								
Quantity	QDAT	QONL	QREA	QINT					
	0								
Schedule Data	CPLX	DSTART	DEND	ISTART	IEND	TSTART	TEND		
	.8	0876	0	0	0	0	0178		
Supplementary Information	YEAR	ESC	TECIMP	MULT	PLTFM	UTIL		TARCST	
	1976	1	1	1.3	1.2	.63			

Notes: -.2 HOL

GC 1610 8/77

RCA

PRICE Software Model Input Worksheet

Filename: STEAF

Page 6 of 7

Title: PROJECT EAF

Application: EARLY WARNING RADAR CONTROL

Date: 2 OCT 79

								Optional	
Descriptors	INST	APPL	RESO	FUNCT	STRU	LEVEL		INTEG	
	54950	0	2.9						
Mix	MDAT	MONL	MREA	MINT	MMAT	MSTR	MOPR	MAPP	APPLE
	0	.82	0	0	0	0	.18		
New Design	DDAT	DONL	DREA	DINT	DMAT	DSTR	DOPR	DAPP	
	.001	.70	.001	.001	.001	.001	.70		
New Code	CDAT	CONL	CREA	CINT	CMAT	CSTR	COPR	CAPP	
	.001	1	.001	.001	.001	.001	1		
Device Types	TDAT	TONL	TREA	TINT					
	0								
Quantity	QDAT	QONL	QREA	QINT					
	0								
Schedule Data	CPLX	DSTART	DEND	ISTART	IEND	TSTART	TEND		
	.9	0476	0	0	0	0	0877		
Supplementary Information	YEAR	ESC	TECIMP	MULT	PLTFM	UTIL		TARCST	
	1976	1	1	1.3	1.2	.75			

Notes: -.2 HOL

GC 1610 8/77

RCA

PRICE Software Model Input Worksheet

Filename: STEAG

Page 1 of 1

Title: PROJECT EAG

Application: EARLY WARNING RADAR SIGNAL PROCESSOR

Date: 2 OCT 79

Optional

Descriptors

INST	APPL	RESO	FUNCT	STRU	LEVEL	INTEG (Optional)
7240	0	2.9				

Mix

MDAT	MONL	MREA	MINT	MMAT	MSTR	MOPR	MAPP (Optional)	APPL8 (Optional)
0	0	1	0	0	0	0		

New Design

DDAT	DONL	DREA	DINT	DMAT	DSTR	DOPR	DAPP (Optional)
.001	.001	1	.001	.001	.001	.001	

New Code

CDAT	CONL	CREA	CINT	CMAT	CSTR	COPR	CAPP (Optional)
.001	.001	1	.001	.001	.001	.001	

Device Types

TDAT	TONL	TREA	TINT
0			

Quantity

QDAT	QONL	QREA	QINT
0			

Schedule Data

CPLX	DSTART	DEND	ISTART	IEND	TSTART	TEND
1.2	0776	0	0	0	0	0877

Supplementary Information

YEAR	ESC	TECIMP	MULT	PLTFM	UTIL	TARCST (Optional)
1976	1	1	1.3	1.2	.75	

Notes:

GC 1610 8/77

RCA

PRICE Software Model Input Worksheet

Filename: STEBA

Page 1 of 5

Title: PROJECT EBA

Application: AIR TRAFFIC CONTROL OPERATIONAL

Date: 7 OCT 79

								Optional	
Descriptors	INST	APPL	RESO	FUNCT	STRU	LEVEL		INTEG	
	40500	0	3.2						
Mix	MDAT	MONL	MREA	MINT	MMAT	MSTR	MOPR	MAPP	APPL8
	0	0	0	0	0	0	1		
New Design	DDAT	DONL	DREA	DINT	DMAT	DSTR	DOPR	DAPP	
	.001	.001	.001	.001	.001	.001	.50		
New Code	CDAT	CONL	CREA	CINT	CMAT	CSTR	COPR	CAPP	
	.001	.001	.001	.001	.001	.001	1		
Device Types	TDAT	TONL	TREA	TINT					
	0								
Quantity	QDAT	QONL	QREA	QINT					
	0								
Schedule Data	CPLX	DSTART	DEND	ISTART	IEND	TSTART	TEND		
	1.5	0173	0	0	0	0	0676		
Supplementary Information	YEAR	ESC	TECIMP	MULT	PLTFM	UTIL		TARCST	
	1973	1	1	1.3	1.2	.75			

Notes:

+.2 MANY NEW HIRES

+.3 CHG REQUIREMENTS

GC 1610 8/77

RCA

PRICE Software Model Input Worksheet

Filename: STEBB

Page 2 of 5

Title: PROJECT EBB

Application: AIR TRAFFIC CONTROL UTILITY

Date: 7 OCT 79

								Optional	
Descriptors	INST	APPL	RESO	FUNCT	STRU	LEVEL		INTEG	
	60000	0	3.2						
Mix	MDAT	MONL	MREA	MINT	MMAT	MSTR	MOPR	MAPP	APPL8
	0	1	0	0	0	0	0		
New Design	DDAT	DONL	DREA	DINT	DMAT	DSTR	DOPR	DAPP	
	.001	.50	.001	.001	.001	.001	.001		
New Code	CDAT	CONL	CREA	CINT	CMAT	CSTR	COPR	CAPP	
	.001	1	.001	.001	.001	.001	.001		
Device Types	TDAT	TONL	TREA	TINT					
	0								
Quantity	QDAT	QONL	QREA	QINT					
	0								
Schedule Data	CPLX	DSTART	DEND	ISTART	IEND	TSTART	TEND		
	1.5	0173	0	0	0	0	0676		
Supplementary Information	YEAR	ESC	TECIMP	MULT	PLTFM	UTIL		TARCST	
	1973	1	1	1.3	1.2	.50			

Notes:

GC 1610 8/77

RCA

PRICE Software Model Input Worksheet

Filename: STEBC

Page 3 of 5

Title: PROJECT EBC

Application: AIR TRAFFIC CONTROL DATA REDUCTION

Date: 7 OCT 79

								Optional	
Descriptors	INST	APPL	RESO	FUNCT	STRU	LEVEL		INTEG	
	18000	0	3.2						
Mix	MDAT	MONL	MREA	MINT	MMAT	MSTR	MOPR	MAPP	APPL8
	0	0	0	0	0	1	0		
New Design	DDAT	DONL	DREA	DINT	DMAT	DSTR	DOPR	DAPP	
	.001	.001	.001	.001	.001	.50	.001		
New Code	CDAT	CONL	CREA	CINT	CMAT	CSTR	COPR	CAPP	
	.001	.001	.001	.001	.001	1	.001		
Device Types	TDAT	TONL	TREA	TINT					
	0								
Quantity	QDAT	QONL	QREA	QINT					
	0								
Schedule Data	CPLX	DSTART	DEND	ISTART	IEND	TSTART	TEND		
	1.5	0173	0	0	0	0	0676		
Supplementary Information	YEAR	ESC	TECIMP	MULT	PLTFM	UTIL		TARCST	
	1973	1	1	1.3	1.2	.75			

Notes:

GC 1610 8/77

RCA

PRICE Software Model Input Worksheet

Filename: STEBD

Page 4 of 5

Title PROJECT EBD

Application AIR TRAFFIC CONTROL SIMULATION

Date 7 OCT 79

								Optional	
Descriptors	INST	APPL	RESO	FUNCT	STRU	LEVEL		INTEG	
	15000	0	3.2						
Mix	MDAT	MONL	MREA	MINT	MMAT	MSTR	MOPR	MAPP	APPLS
	0	0	1	0	0	0	0		
New Design	DDAT	DONL	DREA	DINT	DMAT	DSTR	DOPR	DAPP	
	.001	.001	.50	.001	.001	.001	.001		
New Code	CDAT	CONL	CREA	CINT	CMAT	CSTR	COPR	CAPP	
	.001	.001	1	.001	.001	.001	.001		
Device Types	TDAT	TONL	TREA	TINT					
	0								
Quantity	QDAT	QONL	QREA	QINT					
	0								
Schedule Data	CPLX	DSTART	DEND	ISTART	IEND	TSTART	TEND		
	1.5	0173	0	0	0	0	0676		
Supplementary Information	YEAR	ESC	TECIMP	MULT	PLTFM	UTIL		TARCST	
	1976	1	1	1.3	1.2	.75			

Notes:

GC 1610 8/77

RCA

PRICE Software Model Input Worksheet

Filename: STEBE

Title: PROJECT EBE

Application: AIR TRAFFIC CONTROL TEST

Date: 7 OCT 79

								Optional	
Descriptors	INST 18000	APPL 0	RESO 3.2	FUNCT	STRU	LEVEL		INTEG	
Mix	MDAT 0	MONL 0	MREA 1	MINT 0	MMAT 0	MSTR 0	MOPR 0	MAPP	APPL8
New Design	DDAT .001	DONL .001	DREA .50	DINT .001	DMAT .001	DSTR .001	DOPR .001	DAPP	
New Code	CDAT .001	CONL .001	CREA 1	CINT .001	CMAT .001	CSTR .001	COPR .001	CAPP	
Device Types	TDAT 0	TONL	TREA	TINT					
Quantity	QDAT 0	QONL	QREA	QINT					
Schedule Data	CPLX 1.5	DSTART 0173	DEND 0	ISTART 0	IEND 0	TSTART 0	TEND 0676		
Supplementary Information	YEAR 1973	ESC 1	TECIMP 1	MULT 1.3	PLTFM 1.2	UTIL .75		TARCST	

Notes:

GC 1610 8/77

RCA

PRICE Software Model Input Worksheet

Filename: STECA

Page 1 of 2

Title: PROJECT ECA

Application: TACTICAL COMM SYSTEM APPLICATION

Date: 7 OCT 79

								Optional	
Descriptors	INST	APPL	RESO	FUNCT	STRU	LEVEL		INTEG	
	250000	0	2.9						
Mix	MDAT	MONL	MREA	MINT	MMAT	MSTR	MOPR	MAPP	APPLS
	0	0	1	0	0	0	0		
New Design	DDAT	DONL	DREA	DINT	DMAT	DSTR	DOPR	DAPP	
	.001	.001	1	.001	.001	.001	.001		
New Code	CDAT	CONL	CREA	CINT	CMAT	CSTR	COPR	CAPP	
	.001	.001	1	.001	.001	.001	.001		
Device Types	TDAT	TONL	TREA	TINT					
	0								
Quantity	QDAT	QONL	QREA	QINT					
	0								
Schedule Data	CPLX	DSTART	DEND	ISTART	IEND	TSTART	TEND		
	1.3	0474	0	0	0	0	0277		
Supplementary Information	YEAR	ESC	TECIMP	MULT	PLTFM	UTIL		TARCST	
	1974	1	1	1.3	1.4	.50			

Notes:

+.1 NEW HIRES

+.3 CHG REQUIREMENTS

-.1 HOL.

GC 1610 8/77

RCA

PRICE Software Model Input Worksheet

Filename: STECB

Page 2 of 2

Title: PROJECT ECB

Application: TACTICAL COMM SYSTEM SUPPORT

Date: 7 OCT 79

Optional

Descriptors	INST	APPL	RESO	FUNCT	STRU	LEVEL		INTEG	
	30000	0	2.9						
Mix	MDAT	MONL	MREA	MINT	MMAT	MSTR	MOPR	MAPP	APPL8
	0	1	0	0	0	0	0		
New Design	DDAT	DONL	DREA	DINT	DMAT	DSTR	DOPR	DAPP	
	.001	1	.001	.001	.001	.001	.001		
New Code	CDAT	CONL	CREA	CINT	CMAT	CSTR	COPR	CAPP	
	.001	1	.001	.001	.001	.001	.001		
Device Types	TDAT	TONL	TREA	TINT					
	0								
Quantity	QDAT	QONL	QREA	QINT					
	0								
Schedule Data	CPLX	DSTART	DEND	ISTART	IEND	TSTART	TEND		
	1.3	0474	0	0	0	0	0277		
Supplementary Information	YEAR	ESC	TECIMP	MULT	PLTFM	UTIL		TARCST	
	1974	1	1	1.3	1.4	.50			

Notes:

GC 1610 8/77

RCA

PRICE Software Model Input Worksheet

Filename: FSTCA

Page 1 of 1

Title: PROJECT CA

Application: ATTACK WARNING SYSTEM

Date: 14 OCT 79

								Optional	
Descriptors	INST	APPL	RESO	FUNCT	STRU	LEVEL		INTEG	
	2750000	8.46	3.0						
Mix	MDAT	MONL	MREA	MINT	MMAT	MSTR	MOPR	MAPP	APPLE
	0								
New Design	DDAT	DONL	DREA	DINT	DMAT	DSTR	DOPR	DAPP	
	0								
New Code	CDAT	CONL	CREA	CINT	CMAT	CSTR	COPR	CAPP	
	0								
Device Types	TDAT	TONL	TREA	TINT					
	0								
Quantity	QDAT	QONL	QREA	QINT					
	0								
Schedule Data	CPLX	DSTART	DEND	ISTART	IEND	TSTART	TEND		
	1.1	0172	0	0	0	0	1279		
Supplementary Information	YEAR	ESC	TECIMP	MULT	PLTFM	UTIL		TARCST	
	1972	1	1	1.3	1.2	.50			

Notes:

GC 1610 8/77

RCA

PRICE Software Model Input Worksheet

Filename: FSTCB

Page 1 of 1

Title: PROJECT CB

Application: GROUND BASED RADAR

Date: 14 OCT 79

								Optional	
Descriptors	INST	APPL	RESO	FUNCT	STRU	LEVEL		INTEG	
	25000	8.46	3.0						
Mix	MDAT	MONL	MREA	MINT	MMAT	MSTR	MOPR	MAPP	APPL:
	0								
New Design	DDAT	DONL	DREA	DINT	DMAT	DSTR	DOPR	DAPP	
	0								
New Code	CDAT	CONL	CREA	CINT	CMAT	CSTR	COPR	CAPP	
	0								
Device Types	TDAT	TONL	TREA	TINT					
	0								
Quantity	QDAT	QONL	QREA	QINT					
	0								
Schedule Data	CPLX	DSTART	DEND	ISTART	IEND	TSTART	TEND		
	1.3	0175	0	0	0	0	0676		
Supplementary Information	YEAR	ESC	TECIMP	MULT	PLTFM	UTIL		TARCST	
	1975	1	1	1.3	1.2	.75			

Notes:

GC 1610 8/77

RCA

PRICE Software Model
Input Worksheet

Filename: FSTCC
Page 1 of 1

Title: PROJECT CC
Application: INTELLIGENCE PROCESSING SYSTEM
Date: 14 OCT 79

								Optional	
Descriptors	INST	APPL	RESO	FUNCT	STRU	LEVEL		INTEG	
	250000	8.46	2.9						
Mix	MDAT	MONL	MREA	MINT	MMAT	MSTR	MOPR	MAPP	APPL
	0								
New Design	DDAT	DONL	DREA	DINT	DMAT	DSTR	DOPR	DAPP	
	0								
New Code	CDAT	CONL	CREA	CINT	CMAT	CSTR	COPR	CAPP	
	0								
Device Types	TDAT	TONL	TREA	TINT					
	0								
Quantity	QDAT	QONL	QREA	QINT					
	0								
Schedule Data	CPLX	DSTART	DEND	ISTART	IEND	TSTART	TEND		
	1.2	0175	0	0	0	0	0677		
Supplementary Information	YEAR	ESC	TECIMP	MULT	PLTFM	UTIL		TARCST	
	1975	1	1	1.3	1.2	.50			

Notes:

GC 1610 8/77

RCA

PRICE Software Model Input Worksheet

Filename: FSTCD

Page ___ of ___

Title: PROJECT CD

Application: SEA SURVEILLANCE SYSTEM

Date: 14 OCT 79

								Optional	
Descriptors	INST	APPL	RESO	FUNCT	STRU	LEVEL		INTEG	
	140000	8.46	3.1						
Mix	MDAT	MONL	MREA	MINT	MMAT	MSTR	MOPR	MAPP	APPLE
	0								
New Design	DDAT	DONL	DREA	DINT	DMAT	DSTR	DOPR	DAPP	
	0								
New Code	CDAT	CONL	CREA	CINT	CMAT	CSTR	COPR	CAPP	
	0								
Device Types	TDAT	TONL	TREA	TINT					
	0								
Quantity	QDAT	QONL	QREA	QINT					
	0								
Schedule Data	CPLX	DSTART	DEND	ISTART	IEND	TSTART	TEND		
	1.0	0175	0	0	0	0	1276		
Supplementary Information	YEAR	ESC	TECIMP	MULT	PLTFM	UTIL		TARCST	
	1975	1	1	1.3	1.4	.50			

Notes:

GC 1610 8/77

RCA

PRICE Software Model Input Worksheet

Filename: FSTCE

Page 1 of 1

Title: PROJECT CE

Application: MISSILE C AND C SYSTEM

Date: 14 OCT 79

								Optional	
Descriptors	INST	APPL	RESO	FUNCT	STRU	LEVEL		INTEG	
	11000	8.46	3.1						
Mix	MDAT	MONL	MREA	MINT	MMAT	MSTR	MOPR	MAPP	APPL
	0								
New Design	DDAT	DONL	DREA	DINT	DMAT	DSTR	DOPR	DAPP	
	0								
New Code	CDAT	CONL	CREA	CINT	CMAT	CSTR	COPR	CAPP	
	0								
Device Types	TDAT	TONL	TREA	TINT					
	0								
Quantity	QDAT	QONL	QREA	QINT					
	0								
Schedule Data	CPLX	DSTART	DEND	ISTART	IEND	TSTART	TEND		
	1.3	0175	0	0	0	0	1276		
Supplementary Information	YEAR	ESC	TECIMP	MULT	PLTFM	UTIL		TARCST	
	1975	1	1	1.3	1.4	.75			

Notes:

GC 1610 8/77

RCA

PRICE Software Model Input Worksheet

Filename: FSTCF

Page 1 of 1

Title: PROJECT CF

Application: STRATEGIC COMMUNICATIONS SYSTEM

Date: 14 OCT 79

								Optional	
Descriptors	INST 214000	APPL 8.46	RESO 3.0	FUNCT	STRU	LEVEL		INTEG	
Mix	MDAT 0	MONL	MREA	MINT	MMAT	MSTR	MOPR	MAPP	APPLE
New Design	DDAT 0	DONL	DREA	DINT	DMAT	DSTR	DOPR	DAPP	
New Code	CDAT 0	CONL	CREA	CINT	CMAT	CSTR	COPR	CAPP	
Device Types	TDAT 0	TONL	TREA	TINT					
Quantity	QDAT 0	QONL	QREA	QINT					
Schedule Data	CPLX 1.2	DSTART 0175	DEND 0	ISTART 0	IEND 0	TSTART 0	TEND 1277		
Supplementary Information	YEAR 1975	ESC 1	TECIMP 1	MULT 1.3	PLTFM 1.2	UTIL .50		TARCST	

Notes:

GC 1610 8/77

RCA

PRICE Software Model Input Worksheet

Filename: STASA

Page 1 of 1

Title: PROJECT ASA

Application: INERTIAL NAVIGATION SYSTEM

Date: 16 OCT 79

								Optional	
Descriptors	INST 18000	APPL 0	RESO 2.8	FUNCT	STRU	LEVEL		INTEG	
Mix	MDAT .07	MONL 0	MREA .17	MINT .07	MMAT .33	MSTR .03	MOPR .33	MAPP	APPLE
New Design	DDAT .001	DONL .001	DREA 1	DINT 1	DMAT .6	DSTR .001	DOPR 1	DAPP	
New Code	CDAT 1	CONL .001	CREA 1	CINT 1	CMAT 1	CSTR 1	COPR 1	CAPP	
Device Types	TDAT 0	TONL	TREA	TINT					
Quantity	QDAT 0	QONL	QREA	QINT					
Schedule Data	CPLX 1.0	DSTART 0573	DEND 0	ISTART 0	IEND 0	TSTART 0	TEND 1275		
Supplementary Information	YEAR 1973	ESC 1	TECIMP 0	MULT 1	PLTFM 1.7	UTIL .5		TARCST (ans)	

Notes:

GC 1610 8/77

MRM

PRICE Software Model Input Worksheet

Filename: STASB

Page 1 of 1

Title: PROJECT ASB

Application: NAVIGATION SYSTEM SIMULATION

Date: 16 OCT 79

								Optional	
Descriptors	INST	APPL	RESO	FUNCT	STRU	LEVEL		INTEG	
	2600	0	2.9						
Mix	MDAT	MONL	MREA	MINT	MMAT	MSTR	MOPR	MAPP	APPLE
	0	0	0	.05	.95	0	0		
New Design	DDAT	DONL	DREA	DINT	DMAT	DSTR	DOPR	DAPP	
	.001	.001	.001	1	.1	.001	.001		
New Code	CDAT	CONL	CREA	CINT	CMAT	CSTR	COPR	CAPP	
	.001	.001	.001	1	1	.001	.001		
Device Types	TDAT	TONL	TREA	TINT					
	0								
Quantity	QDAT	QONL	QREA	QINT					
	0								
Schedule Data	CPLX	DSTART	DEND	ISTART	IEND	TSTART	TEND		
	.9	1177	0	0	0	0	0278		
Supplementary Information	YEAR	ESC	TECIMP	MULT	PLTFM	UTIL		TARCST	
	1977	1	0	1	1.0	.3		(12)	

Notes:

GC 1610 8/77

PRICE Software Model Input Worksheet

Filename: STASC

Page 1 of 1

Title: PROJECT ASC

Application: AIRBORNE RADAR SYSTEM

Date: 16 OCT 79

								Optional	
Descriptors	INST	APPL	RESO	FUNCT	STRU	LEVEL		INTEG	
	73750	0	2.7						
Mix	MDAT	MONL	MREA	MINT	MMAT	MSTR	MOPR	MAPP	APPL8
	.15	0	.15	.05	.65	0	0		
New Design	DDAT	DONL	DREA	DINT	DMAT	DSTR	DOPR	DAPP	
	1	.001	1	1	.99	.001	.001		
New Code	CDAT	CONL	CREA	CINT	CMAT	CSTR	COPR	CAPP	
	1	.001	1	1	.99	.001	.001		
Device Types	TDAT	TONL	TREA	TINT					
	0								
Quantity	QDAT	QONL	QREA	QINT					
	0								
Schedule Data	CPLX	DSTART	DEND	ISTART	IEND	TSTART	TEND		
	.9	0574	0	0	0	0	0579		
Supplementary Information	YEAR	ESC	TECIMP	MULT	PLTFM	UTIL		TARCST	
	1974	1	0	1.12	1.7	.85		(2000)	

Notes:

GC 1610 8/77

RCA

PRICE Software Model Input Worksheet

Filename: STDCA

Page 1 of 1

Title: PROJECT DCA

Application: OPERATIONAL PLANNING SYSTEM

Date: 16 OCT 79

								Optional	
Descriptors	INST	APPL	RESO	FUNCT	STRU	LEVEL		INTEG	
	315000	1.588	1.5						
Mix	MDAT	MONL	MREA	MINT	MMAT	MSTR	MOPR	MAPP	APPLE
	0								
New Design	DDAT	DONL	DREA	DINT	DMAT	DSTR	DOPR	DAPP	
	0								
New Code	CDAT	CONL	CREA	CINT	CMAT	CSTR	COPR	CAPP	
	0								
Device Types	TDAT	TONL	TREA	TINT					
	0								
Quantity	QDAT	QONL	QREA	QINT					
	0								
Schedule Data	CPLX	DSTART	DEND	ISTART	IEND	TSTART	TEND		
	1.0	0278	0	0	0	0	0382		
Supplementary Information	YEAR	ESC	TECIMP	MULT	PLTFM	UTIL		TARCST	
	1978	1	1	1	.8	.1		(406)	

Notes:

GC 1610 8/77

RCA

PRICE Software Model Input Worksheet

Filename: STDCB

Page 1 of 1

Title: PROJECT DCB

Application: ENGINEERING MANAGEMENT SYSTEM

Date: 16 OCT 79

Descriptors	INST	APPL	RESO	FUNCT	STRU	LEVEL		Optional: INTEG	
	66000	1.588	1.5						

Mix	MDAT	MONL	MREA	MINT	MMAT	MSTR	MOPR	MAPP	APPLE
	0								

New Design	DDAT	DONL	DREA	DINT	DMAT	DSTR	DOPR	DAPP
	0							

New Code	CDAT	CONL	CREA	CINT	CMAT	CSTR	COPR	CAPP
	0							

Device Types	TDAT	TONL	TREA	TINT
	0			

Quantity	QDAT	QONL	QREA	QINT
	0			

Schedule Data	CPLX	DSTART	DEND	ISTART	IEND	TSTART	TEND
	1.0	0873 0175	0	0	0	0	1276

Supplementary Information	YEAR	ESC	TECIMP	MULT	PLTFM	UTIL	TARCST
	1973 1975	1	1	1	.8	.1	(62)

Notes:

GC 1610 8/77

RCA

PRICE Software Model Input Worksheet

Filename: STALA

Page 1 of 1

Title: PROJECT ALA

Application: MODIFICATION MANAGEMENT SYSTEM

Date: 16 OCT 79

								Optional	
Descriptors	INST 27900	APPL 1.588	RESO 1.5	FUNCT	STRU	LEVEL		INTEG .	
Mix	MDAT 0	MONL	MREA	MINT	MMAT	MSTR	MOPR	MAPP	APP
New Design	DDAT 0	DONL	DREA	DINT	DMAT	DSTR	DOPR	DAPP	
New Code	CDAT 0	CONL	CREA	CINT	CMAT	CSTR	COPR	CAPP	
Device Types	TDAT 0	TONL	TREA	TINT					
Quantity	QDAT 0	QONL	QREA	QINT					
Schedule Data	CPLX 1.0	DSTART 0578	DEND 0	ISTART 0	IEND 0	TSTART 0	TEND 0179		
Supplementary Information	YEAR 1978	ESC 1	TECIMP 1	MULT 1	PLTFM .8	UTIL .1		TARCST [illegible]	

Notes:

GC 1610 8/77

PRICE Software Model Input Worksheet

Filename: STALB

Page 1 of 1

Title: PROJECT ALB

Application: MATERIAL REPORTING SYSTEM

Date: 16 OCT 79

								Optional	
Descriptors	INST	APPL	RESO	FUNCT	STRU	LEVEL		INTEG	
	70140	1.588	1.5						
Mix	MDAT	MONL	MREA	MINT	MMAT	MSTR	MOPR	MAPP	APPLE
	0								
New Design	DDAT	DONL	DREA	DINT	DMAT	DSTR	DOPR	DAPP	
	0								
New Code	CDAT	CONL	CREA	CINT	CMAT	CSTR	COPR	CAPP	
	0								
Device Types	TDAT	TONL	TREA	TINT					
	0								
Quantity	QDAT	QONL	QREA	QINT					
	0								
Schedule Data	CPLX	DSTART	DEND	ISTART	IEND	TSTART	TEND		
	1.0	1175	0	0	0	0	0878		
Supplementary Information	YEAR	ESC	TECIMP	MULT	PLTFM	UTIL		TARCST	
	1975	1	1	1	.8	.1		(112)	

Notes:

GC 1610 8/77

RCA

PRICE Software Model Input Worksheet

Filename: STAUA

Page 1 of 1

Title: PROJECT AUA

Application: WARTIME SIMULATION MODEL

Date: 16 OCT 79

								Optional	
Descriptors	INST	APPL	RESO	FUNCT	STRU	LEVEL		INTEG	
	30000	1.588	1.5						
Mix	MDAT	MONL	MREA	MINT	MMAT	MSTR	MOPR	MAPP	APPLE
	0								
New Design	DDAT	DONL	DREA	DINT	DMAT	DSTR	DOPR	DAPP	
	0								
New Code	CDAT	CONL	CREA	CINT	CMAT	CSTR	COPR	CAPP	
	0								
Device Types	TDAT	TONL	TREA	TINT					
	0								
Quantity	QDAT	QONL	QREA	QINT					
	0								
Schedule Data	CPLX	DSTART	DEND	ISTART	IEND	TSTART	TEND		
	1.0	1077	0	0	0	0	0878		
Supplementary Information	YEAR	ESC	TECIMP	MULT	PLTFM	UTIL		TARCST	
	1977	1	1	1	.8	.1		(77)	

Notes:

GC 1610 8/77

RCA

PRICE Software Model Input Worksheet

Filename: STAUB

Page 1 of 1

Title: PROJECT AUB

Application: MAINTENANCE SIMULATION SYSTEM

Date: 16 OCT 79

								Optional	
Descriptors	INST	APPL	RESO	FUNCT	STRU	LEVEL		INTEG	
	2900	1.588	1.5						
Mix	MDAT	MONL	MREA	MINT	MMAT	MSTR	MOPR	MAPP	APPL8
	0								
New Design	DDAT	DONL	DREA	DINT	DMAT	DSTR	DOPR	DAPP	
	0								
New Code	CDAT	CONL	CREA	CINT	CMAT	CSTR	COPR	CAPP	
	0								
Device Types	TDAT	TONL	TREA	TINT					
	0								
Quantity	QDAT	QONL	QREA	QINT					
	0								
Schedule Data	CPLX	DSTART	DEND	ISTART	IEND	TSTART	TEND		
	1.0	0278	0	0	0	0	0778		
Supplementary Information	YEAR	ESC	TECIMP	MULT	PLTFM	UTIL		TARCST	
	1978	1	1	1	.8	.1		(8)	

Notes:

GC 1610 8/77

RCA

APPENDIX B

PRICE-S Model Outputs

--- PRICE SOFTWARE MODEL ---

DATE 13-OCT-79 TIME 21:18

PROJECT EWR — EARLY WARNING RADAR OPERATION

INPUT DATA

FILENAME: STEWR — DATED: 2 OCT 79

DESCRIPTORS

INSTRUCTIONS	25500	APPLICATION	0.000	RESOURCE	2.900
FUNCTIONS	0	STRUCTURE	0.000	LEVEL	0.000

APPLICATION CATEGORIES		NEW DEVELOPMENT		SYSTEM CONFIGURATION	
	MIX	DESIGN	CODE	TYPES	QUANTITY
DATA S/R	0.00	0.00	0.00	0	0
ONLINE COMM	0.00	0.00	0.00	0	0
REALTIME C&C	0.00	0.00	0.00	0	0
INTERACTIVE	0.00	0.00	0.00	0	0
MATHEMATICAL	0.00	0.00	0.00	***	***
STRING MANIP	0.00	0.00	0.00	***	***
OPR SYSTEMS	1.00	1.00	1.00	***	***

SCHEDULE

COMPLEXITY	0.900				
DESIGN START	APR 76	IMPL START	0	T&I START	0
DESIGN END	0	IMPL END	0	T&I END	NOV 77

SUPPLEMENTAL INFORMATION

YEAR	1976	ESCALATION	1.000	TECH IMP	1.00
MULTIPLIER	1.300	PLATFORM	1.2	UTILIZATION	0.63

PROGRAM COSTS

COST ELEMENTS	DESIGN	IMPL	T & I	TOTAL
SYSTEMS ENGINEERING	207.	9.	180.	396.
PROGRAMMING	57.	54.	74.	184.
CONFIGURATION CONTROL	24.	12.	100.	136.
DOCUMENTATION	21.	3.	36.	60.
PROGRAM MANAGEMENT	26.	3.	19.	48.
TOTAL	335.	82.	408.	825.

ADDITIONAL DATA

DESCRIPTORS

INSTRUCTIONS	25500	APPLICATION	10.952	RESOURCE	2.900
FUNCTIONS	283	STRUCTURE	0.000	LEVEL	0.000

SCHEDULE

COMPLEXITY	0.900				
DESIGN START	APR 76	IMPL START	0	T&I START	0
DESIGN END	0	IMPL END	0	T&I END	NOV 77

--- PRICE SOFTWARE MODEL ---

DATE 13-OCT-79 TIME 21:19

PROJECT EWR EARLY WARNING RADAR OPERATION

SENSITIVITY DATA

COMPLEXITY

RESOURCE		.800		.900		1.000
2.800	COST	755.	COST	783.	COST	825.
	MONTHS	19.0	MONTHS	19.0	MONTHS	19.0
2.900	COST	795.	COST	825.	COST	870.
	MONTHS	19.0	MONTHS	19.0	MONTHS	19.0
3.000	COST	835.	COST	862.	COST	916.
	MONTHS	19.0	MONTHS	19.0	MONTHS	19.0

SCHEDULE EFFECT SUMMARY

ACTIVITY LENGTH IN MONTHS

COMPLEXITY = 0.900	DESIGN		IMPL		T & I	TOTAL
SPECIFIED SCHEDULE	0.0		0.0		0.0	19.0
(OVERLAP)		(0.0%)		(0.0%)		
TYPICAL SCHEDULE	5.8		5.3		2.5	13.9
(OVERLAP)		(2.4%)		(2.2%)		

DEVELOPMENT COSTS

COMPLEXITY = 0.900	DESIGN	IMPL	T & I	TOTAL
SPECIFIED SCHEDULE	335.	82.	408.	825.
TYPICAL SCHEDULE	315.	77.	383.	775.
ESTIMATED PENALTY	20.	5.	25.	50.

--- PRICE SOFTWARE MODEL ---

DATE 13-OCT-79 TIME 21:22

PROJECT EWR — EARLY WARNING RADAR APPLICATION

INPUT DATA

FILENAME: STEWR — DATED: 2 OCT 79

DESCRIPTORS

INSTRUCTIONS	180500	APPLICATION	0.000	RESOURCE	2.900
FUNCTIONS	0	STRUCTURE	0.000	LEVEL	0.000

APPLICATION CATEGORIES		NEW DEVELOPMENT		SYSTEM CONFIGURATION	
	MIX	DESIGN	CODE	TYPES	QUANTITY
DATA S/R	0.00	0.00	0.00	0	0
ONLINE COMM	0.00	0.00	0.00	0	0
REALTIME C&C	0.40	0.90	1.00	0	0
INTERACTIVE	0.44	0.90	1.00	0	0
MATHEMATICAL	0.00	0.00	0.00	***	***
STRING MANIP	0.07	0.90	1.00	***	***
OPR SYSTEMS	0.09	0.90	1.00	***	***

SCHEDULE

COMPLEXITY	1.000				
DESIGN START	APR 76	IMPL START	0	T&I START	0
DESIGN END	0	IMPL END	0	T&I END	DEC 77

SUPPLEMENTAL INFORMATION

YEAR	1976	ESCALATION	1.000	TECH IMP	1.00
MULTIPLIER	1.300	PLATFORM	1.2	UTILIZATION	0.63

PROGRAM COSTS

COST ELEMENTS	DESIGN	IMPL	T & I	TOTAL
SYSTEMS ENGINEERING	1452.	89.	1403.	2945.
PROGRAMMING	314.	494.	574.	1381.
CONFIGURATION CONTROL	243.	143.	935.	1321.
DOCUMENTATION	198.	40.	353.	592.
PROGRAM MANAGEMENT	193.	43.	190.	426.
TOTAL	2400.	809.	3456.	6665.

ADDITIONAL DATA

DESCRIPTORS

INSTRUCTIONS	180500	APPLICATION	9.350	RESOURCE	2.900
FUNCTIONS	2006	STRUCTURE	0.000	LEVEL	0.000

SCHEDULE

COMPLEXITY	1.000				
DESIGN START	APR 76	IMPL START	0	T&I START	0
DESIGN END	0	IMPL END	0	T&I END	DEC 77

--- PRICE SOFTWARE MODEL ---

DATE 13-OCT-79   TIME 21:27

PROJECT EWR   EARLY WARNING RADAR APPLICATION

SENSITIVITY DATA

COMPLEXITY

RESOURCE		.900		1.000		1.100	
2.800	COST	4954.	COST	6232.	COST	7835.
	MONTHS	20.0	MONTHS	20.0	MONTHS	20.0
2.900	COST	5288.	COST	6665.	COST	8378.
	MONTHS	20.0	MONTHS	20.0	MONTHS	20.0
3.000	COST	5631.	COST	7102.	COST	8938.
	MONTHS	20.0	MONTHS	20.0	MONTHS	20.0

SCHEDULE EFFECT SUMMARY

ACTIVITY LENGTH IN MONTHS

COMPLEXITY = 1.000	DESIGN		IMPL		T & I	TOTAL
SPECIFIED SCHEDULE	0.0		0.0		0.0	20.0
(OVERLAP)		(0.0%)		(0.0%)		
TYPICAL SCHEDULE	12.5		11.9		16.8	30.1
(OVERLAP)		(5.2%)		(5.4%)		

DEVELOPMENT COSTS

COMPLEXITY = 1.000	DESIGN	IMPL	T & I	TOTAL
SPECIFIED SCHEDULE	2400.	809.	3456.	6665.
TYPICAL SCHEDULE	1828.	510.	2664.	5002.
ESTIMATED PENALTY	573.	298.	792.	1663.

--- PRICE SOFTWARE MODEL ---

DATE 13-OCT-79 TIME 21:27

PROJECT EAC EARLY WARNING RADAR SIMULATION

INPUT DATA

FILENAME: STEAC DATED: 2 OCT 79

DESCRIPTORS					
INSTRUCTIONS	70200	APPLICATION	0.000	RESOURCE	2.900
FUNCTIONS	0	STRUCTURE	0.000	LEVEL	0.000

APPLICATION CATEGORIES		NEW DEVELOPMENT		SYSTEM CONFIGURATION	
	MIX	DESIGN	CODE	TYPES	QUANTITY
DATA S/R	0.00	0.00	0.00	0	0
ONLINE COMM	0.00	0.00	0.00	0	0
REALTIME C&C	0.00	0.00	0.00	0	0
INTERACTIVE	0.30	0.40	1.00	0	0
MATHEMATICAL	0.00	0.00	0.00	***	***
STRING MANIP	0.70	0.40	1.00	***	***
OPR SYSTEMS	0.00	0.00	0.00	***	***

SCHEDULE					
COMPLEXITY	1.000				
DESIGN START	AUG 76	IMPL START	0	T&T START	0
DESIGN END	0	IMPL END	0	T&T END	OCT 77

SUPPLEMENTAL INFORMATION					
YEAR	1976	ESCALATION	1.000	TECH IMP	1.00
MULTIPLIER	1.300	PLATFORM	1.2	UTILIZATION	0.63

PROGRAM COSTS

COST ELEMENTS	DESIGN	IMPL	T & T	TOTAL
SYSTEMS ENGINEERING	145.	11.	198.	354.
PROGRAMMING	31.	60.	81.	172.
CONFIGURATION CONTROL	22.	16.	121.	159.
DOCUMENTATION	18.	4.	45.	67.
PROGRAM MANAGEMENT	17.	5.	24.	46.
TOTAL	235.	96.	469.	799.

ADDITIONAL DATA

DESCRIPTORS					
INSTRUCTIONS	70200	APPLICATION	4.903	RESOURCE	2.900
FUNCTIONS	280	STRUCTURE	0.000	LEVEL	0.000

SCHEDULE					
COMPLEXITY	1.000				
DESIGN START	AUG 76	IMPL START	0	T&T START	0
DESIGN END	0	IMPL END	0	T&T END	OCT 77

--- PRICE SOFTWARE MODEL ---

DATE 13-OCT-79 TIME 21:28

PROJECT EAL EARLY WARNING RADAR SIMULATION

SENSITIVITY DATA

COMPLEXITY

RESOURCE		.900	1.000	1.100
2.800	COST	688.	756.	868.
	MONTHS	14.0	14.0	14.0
2.900	COST	726.	799.	919.
	MONTHS	14.0	14.0	14.0
3.000	COST	763.	843.	922.
	MONTHS	14.0	14.0	14.0

SCHEDULE EFFECT SUMMARY

ACTIVITY LENGTH IN MONTHS

COMPLEXITY = 1.000	DESIGN	IMPL	T & I	TOTAL
SPECIFIED SCHEDULE	0.0	0.0	0.0	14.0
%OVERLAP	(0.0%)	(0.0%)		
TYPICAL SCHEDULE	5.6	6.0	9.0	15.3
%OVERLAP	(2.5%)	(2.7%)		

DEVELOPMENT COSTS

COMPLEXITY = 1.000	DESIGN	IMPL	T & I	TOTAL
SPECIFIED SCHEDULE	235.	96.	469.	799.
TYPICAL SCHEDULE	231.	93.	464.	788.
ESTIMATED PENALTY	4.	2.	5.	11.

```
                    --- PRICE SOFTWARE MODEL ---

                    DATE 13-OCT-79     TIME 21:32

PROJECT EAD                              EARLY WARNING RADAR PROG TOOLS

                              INPUT DATA
FILENAME: STEAD                                          DATED: 2 OCT 79

DESCRIPTORS
  INSTRUCTIONS  100000       APPLICATION   0.000       RESOURCE      2.900
  FUNCTIONS          0       STRUCTURE     0.000       LEVEL         0.000

APPLICATION CATEGORIES        NEW DEVELOPMENT          SYSTEM CONFIGURATION
                    WTS       DESIGN       CODE        TYPES      QUANTITY
  DATA S/R          0.50      0.50         1.00          0            0
  ONLINE COMM       0.00      0.00         0.00          0            0
  REALTIME C&C      0.00      0.00         0.00          0            0
  INTERACTIVE       0.00      0.00         0.00          0            0
  MATHEMATICAL      0.00      0.00         0.00         ***          ***
  STRING MANIP      0.50      0.50         1.00         ***          ***
  OPR SYSTEMS       0.00      0.00         0.00         ***          ***

SCHEDULE
  COMPLEXITY        0.800
  DESIGN START      APR 76   IMPL START       0        T&T START       0
  DESIGN END          0      IMPL END         0        T&T END      JAN 77

SUPPLEMENTAL INFORMATION
  YEAR              1976     ESCALATION    1.000       TECH IMP      1.00
  MULTIPLIER        1.300    PLATFORM      1.2         UTILIZATION   0.63

                              PROGRAM COSTS

COST ELEMENTS                 DESIGN       IMPL        T & T         TOTAL
  SYSTEMS ENGINEERING          151.         10.         185.          345.
  PROGRAMMING                   54.         62.          76.          191.
  CONFIGURATION CONTROL         16.         15.         117.          148.
  DOCUMENTATION                 17.          4.          43.           64.
  PROGRAM MANAGEMENT            26.          4.          23.           54.
     TOTAL                     263.         95.         444.          802.

                             ADDITIONAL DATA
DESCRIPTORS
  INSTRUCTIONS  100000       APPLICATION   3.208       RESOURCE      2.900
  FUNCTIONS       1111       STRUCTURE     0.000       LEVEL         0.000

SCHEDULE
  COMPLEXITY        0.800
  DESIGN START      APR 76   IMPL START       0        T&T START       0
  DESIGN END          0      IMPL END         0        T&T END      JAN 77
```

--- PRICE SOFTWARE MODEL ---

DATE 13-OCT-79 TIME 21:33

PROJECT EWR EARLY WARNING RADAR PROG TOOLS

SENSITIVITY DATA

COMPLEXITY

RESOURCE		.700		.800		.900	
2.800	COST	625.	COST	757.	COST	942.
	MONTHS	9.0	MONTHS	9.0	MONTHS	9.0
2.900	COST	681.	COST	802.	COST	1005.
	MONTHS	9.0	MONTHS	9.0	MONTHS	9.0
3.000	COST	697.	COST	849.	COST	1064.
	MONTHS	9.0	MONTHS	9.0	MONTHS	9.0

SCHEDULE EFFECT SUMMARY

ACTIVITY LENGTH IN MONTHS

COMPLEXITY = 0.800	DESIGN		IMPL		T & T	TOTAL
SPECIFIED SCHEDULE	0.0		0.0		0.0	9.0
(OVERLAP)		(0.0%)		(0.0%)		
TYPICAL SCHEDULE	4.3		4.4		6.6	12.0
(OVERLAP)		(1.6%)		(1.7%)		

DEVELOPMENT COSTS

COMPLEXITY = 0.800	DESIGN	IMPL	T & T	TOTAL
SPECIFIED SCHEDULE	263.	95.	444.	802.
TYPICAL SCHEDULE	226.	73.	382.	681.
ESTIMATED PENALTY	37.	22.	62.	121.

--- PRICE SOFTWARE MODEL ---

DATE 13-OCT-79 TIME 21:38

PROJECT PAR .... EARLY WARNING RADAR DATA REDUCTION

INPUT DATA

FILENAME: STEAR .... DATED: 2 OCT 79

DESCRIPTORS

INSTRUCTIONS	56500	APPLICATION	0.000	RESOURCE	2.900
FUNCTIONS	0	STRUCTURE	0.000	LEVEL	0.000

APPLICATION CATEGORIES		NEW DEVELOPMENT		SYSTEM CONFIGURATION	
	MIX	DESIGN	CODE	TYPES	QUANTITY
DATA S/R	0.00	0.00	0.00	0	0
ONLINE COMM	0.00	0.00	0.00	0	0
REALTIME C&C	0.00	0.00	0.00	0	0
INTERACTIVE	0.00	0.00	0.00	0	0
MATHEMATICAL	0.00	0.00	0.00	***	***
STRING MANIP	1.00	1.00	1.00	***	***
OPR SYSTEMS	0.00	0.00	0.00	***	***

SCHEDULE

COMPLEXITY	0.800				
DESIGN START	AUG 76	IMPL START	0	T&E START	0
DESIGN END	0	IMPL END	0	T&E END	JAN 78

SUPPLEMENTAL INFORMATION

YEAR	1976	ESCALATION	1.000	TECH IMP	1.00
MULTIPLIER	1.300	PLATFORM	1.2	UTILIZATION	0.63

PROGRAM COSTS

COST ELEMENTS	DESIGN	IMPL	T & E	TOTAL
SYSTEMS ENGINEERING	113.	4.	97.	214.
PROGRAMMING	40.	29.	40.	109.
CONFIGURATION CONTROL	11.	2.	58.	76.
DOCUMENTATION	12.	2.	21.	35.
PROGRAM MANAGEMENT	18.	2.	11.	31.
TOTAL	194.	44.	226.	465.

ADDITIONAL DATA

DESCRIPTORS

INSTRUCTIONS	56500	APPLICATION	2.311	RESOURCE	2.900
FUNCTIONS	628	STRUCTURE	0.000	LEVEL	0.000

SCHEDULE

COMPLEXITY	0.800				
DESIGN START	AUG 76	IMPL START	0	T&E START	0
DESIGN END	0	IMPL END	0	T&E END	JAN 78

--- PRICE SOFTWARE MODEL ---

DATE 13-OCT-79 TIME 21:38

PROJECT EWR EARLY WARNING RADAR DATA REDUCTION

SENSITIVITY DATA

COMPLEXITY

RESOURCE		.700		.800		.900	
2.800	COST	446.	COST	443.	COST	442.	
	MONTHS	12.0	MONTHS	12.0	MONTHS	12.0	
2.900	COST	468.	COST	465.	COST	470.	
	MONTHS	12.0	MONTHS	12.0	MONTHS	12.0	
3.000	COST	490.	COST	482.	COST	492.	
	MONTHS	12.0	MONTHS	12.0	MONTHS	12.0	

SCHEDULE EFFECT SUMMARY

ACTIVITY LENGTH IN MONTHS

COMPLEXITY = 0.800	DESIGN	IMPL	T & I	TOTAL
SPECIFIED SCHEDULE	0.0	0.0	0.0	12.0
(OVERLAP)	(0.0%)	(0.0%)		
TYPICAL SCHEDULE	3.2	3.3	4.8	9.2
(OVERLAP)	(1.4%)	(1.3%)		

DEVELOPMENT COSTS

COMPLEXITY = 0.800	DESIGN	IMPL	T & I	TOTAL
SPECIFIED SCHEDULE	194.	44.	226.	465.
TYPICAL SCHEDULE	169.	38.	195.	402.
ESTIMATED PENALTY	26.	6.	32.	63.

--- PRICE SOFTWARE MODEL ---

DATE 13-OCT-79 TIME 21:41

PROJECT EWR ... EARLY WARNING RADAR CONTROL

INPUT DATA

FILENAME: STEWR ... DATED: 2 OCT 79

DESCRIPTORS

INSTRUCTIONS	54950	APPLICATION	0.000	RESOURCE	2.900
FUNCTIONS	0	STRUCTURE	0.000	LEVEL	0.000

APPLICATION CATEGORIES	MIX	NEW DEVELOPMENT DESIGN	CODE	SYSTEM CONFIGURATION TYPES	QUANTITY
DATA S/R	0.00	0.00	0.00	0	0
ONLINE COMM	0.82	0.20	1.00	0	0
REALTIME C&C	0.00	0.00	0.00	0	0
INTERACTIVE	0.00	0.00	0.00	0	0
MATHEMATICAL	0.00	0.00	0.00	***	***
STRING MANIP	0.00	0.00	0.00	***	***
OPR SYSTEMS	0.18	0.20	1.00	***	***

SCHEDULE

COMPLEXITY	0.900				
DESIGN START	APR 76	IMPL START	0	T&I START	0
DESIGN END	0	IMPL END	0	T&I END	AUG 77

SUPPLEMENTAL INFORMATION

YEAR	1976	ESCALATION	1.000	TECH IMP	1.10
MULTIPLIER	1.300	PLATFORM	1.2	UTILIZATION	0.75

PROGRAM COSTS

COST ELEMENTS	DESIGN	IMPL	T & I	TOTAL
SYSTEMS ENGINEERING	243.	12.	256.	511.
PROGRAMMING	66.	70.	105.	241.
CONFIGURATION CONTROL	30.	17.	153.	200.
DOCUMENTATION	27.	5.	56.	88.
PROGRAM MANAGEMENT	33.	5.	30.	67.
TOTAL	399.	108.	593.	1106.

ADDITIONAL DATA

DESCRIPTORS

INSTRUCTIONS	54950	APPLICATION	7.029	RESOURCE	2.900
FUNCTIONS	611	STRUCTURE	0.000	LEVEL	0.000

SCHEDULE

COMPLEXITY	0.900				
DESIGN START	APR 76	IMPL START	0	T&I START	0
DESIGN END	0	IMPL END	0	T&I END	AUG 77

--- PRICE SOFTWARE MODEL ---

DATE 13-OCT-79 TIME 21:42

PROJECT EWR EARLY WARNING RADAR CONTROL

SENSITIVITY DATA

COMPLEXITY

RESOURCE		.800		.900		1.000
2.800	COST	977.	COST	1048.	COST	1153.
	MONTHS	16.0	MONTHS	16.0	MONTHS	16.0
2.900	COST	1030.	COST	1106.	COST	1220.
	MONTHS	16.0	MONTHS	16.0	MONTHS	16.0
3.000	COST	1083.	COST	1165.	COST	1288.
	MONTHS	16.0	MONTHS	16.0	MONTHS	16.0

SCHEDULE EFFECT SUMMARY

ACTIVITY LENGTH IN MONTHS

COMPLEXITY = 0.900	DESIGN	IMPL	T & I	TOTAL
SPECIFIED SCHEDULE	0.0	0.0	0.0	16.0
%OVERLAP	(0.0%)	(0.0%)		
TYPICAL SCHEDULE	6.0	5.9	8.6	15.4
%OVERLAP	(2.5%)	(2.5%)		

DEVELOPMENT COSTS

COMPLEXITY = 0.900	DESIGN	IMPL	T & I	TOTAL
SPECIFIED SCHEDULE	399.	108.	599.	1106.
TYPICAL SCHEDULE	399.	108.	598.	1105.
ESTIMATED PENALTY	0.	0.	1.	1.

--- PRICE SOFTWARE MODEL ---

DATE 13-OCT-79   TIME 21:45

PROJECT EWS   EARLY WARNING RADAR SIGNAL PROCESSOR

INPUT DATA

FILENAME: SIEWS   DATED: 2 OCT 79

DESCRIPTORS

INSTRUCTIONS	7240	APPLICATION	0.000	RESOURCE	2.900
FUNCTIONS	0	STRUCTURE	0.000	LEVEL	0.000

APPLICATION CATEGORIES		NEW DEVELOPMENT		SYSTEM CONFIGURATION	
	MIX	DESIGN	CODE	TYPES	QUANTITY
DATA S/R	0.00	0.00	0.00	0	0
ONLINE COMM	0.00	0.00	0.00	0	0
REALTIME C&C	1.00	1.00	1.00	0	0
INTERACTIVE	0.00	0.00	0.00	0	0
MATHEMATICAL	0.00	0.00	0.00	***	***
STRING MANIP	0.00	0.00	0.00	***	***
OPR SYSTEMS	0.00	0.00	0.00	***	***

SCHEDULE

COMPLEXITY	1.200				
DESIGN START	JUL 76	IMPL START	0	T&I START	0
DESIGN END	0	IMPL END	0	T&I END	AUG 77

SUPPLEMENTAL INFORMATION

YEAR	1976	ESCALATION	1.000	TECH IMP	1.00
MULTIPLIER	1.300	PLATFORM	1.2	UTILIZATION	0.75

PROGRAM COSTS

COST ELEMENTS	DESIGN	IMPL	T & I	TOTAL
SYSTEMS ENGINEERING	74.	4.	63.	140.
PROGRAMMING	11.	17.	26.	54.
CONFIGURATION CONTROL	12.	4.	30.	46.
DOCUMENTATION	8.	1.	10.	20.
PROGRAM MANAGEMENT	5.	1.	5.	12.
TOTAL	110.	27.	135.	272.

ADDITIONAL DATA

DESCRIPTORS

INSTRUCTIONS	7240	APPLICATION	8.460	RESOURCE	2.900
FUNCTIONS	80	STRUCTURE	0.000	LEVEL	0.000

SCHEDULE

COMPLEXITY	1.200				
DESIGN START	JUL 76	IMPL START	0	T&I START	0
DESIGN END	0	IMPL END	0	T&I END	AUG 77

--- PRICE SOFTWARE MODEL ---

DATE 13-OCT-79 TIME 21:47

PROJECT EWS EARLY WARNING RADAR SIGNAL PROCESSOR

SENSITIVITY DATA

COMPLEXITY

RESOURCE	1.100		1.200		1.300	
2.800	COST	246.	COST	259.	COST	275.
	MONTHS	13.0	MONTHS	13.0	MONTHS	13.0
2.900	COST	258.	COST	272.	COST	289.
	MONTHS	13.0	MONTHS	13.0	MONTHS	13.0
3.000	COST	270.	COST	285.	COST	303.
	MONTHS	13.0	MONTHS	13.0	MONTHS	13.0

SCHEDULE EFFECT SUMMARY

ACTIVITY LENGTH IN MONTHS

COMPLEXITY = 1.200	DESIGN	IMPL	T & T	TOTAL
SPECIFIED SCHEDULE	0.0	0.0	0.0	13.0
(OVERLAP)	(0.0)	(0.0)		
TYPICAL SCHEDULE	4.9	4.7	6.5	11.2
(OVERLAP)	(2.5)	(2.4)		

DEVELOPMENT COSTS

COMPLEXITY = 1.200	DESIGN	IMPL	T & T	TOTAL
SPECIFIED SCHEDULE	110.	27.	135.	272.
TYPICAL SCHEDULE	108.	27.	133.	268.
ESTIMATED PENALTY	2.	0.	2.	4.

--- PRICE SOFTWARE MODEL ---

DATE 19-OCT-79 TIME 00:35

PROJECT FAA — AIR TRAFFIC CONTROL OPERATIONAL

INPUT DATA

FILENAME: STEPA — DATED: 7 OCT 79

DESCRIPTORS

INSTRUCTIONS	40500	APPLICATION	0.000	RESOURCE	3.200
FUNCTIONS	0	STRUCTURE	0.000	LEVEL	0.000

APPLICATION CATEGORIES		NEW DEVELOPMENT		SYSTEM CONFIGURATION	
	MIX	DESIGN	CODE	TYPES	QUANTITY
DATA S/R	0.00	0.00	0.00	0	0
ONLINE COMM	0.00	0.00	0.00	0	0
REALTIME C&C	0.00	0.00	0.00	0	0
INTERACTIVE	0.00	0.00	0.00	0	0
MATHEMATICAL	0.00	0.00	0.00	***	***
STRING MANIP	0.00	0.00	0.00	***	***
OPR SYSTEMS	1.00	0.50	1.00	***	***

SCHEDULE

COMPLEXITY	1.500				
DESIGN START	JAN 73	IMPL START	0	T&T START	0
DESIGN END	0	IMPL END	0	T&T END	JUN 76

SUPPLEMENTAL INFORMATION

YEAR	1973	ESCALATION	1.000	TECH IMP	1.00
MULTIPLIER	1.300	PLATFORM	1.2	UTILIZATION	0.75

PROGRAM COSTS

COST ELEMENTS	DESIGN	IMPL	T & I	TOTAL
SYSTEMS ENGINEERING	370.	30.	482.	882.
PROGRAMMING	32.	124.	197.	354.
CONFIGURATION CONTROL	110.	38.	279.	427.
DOCUMENTATION	66.	11.	101.	178.
PROGRAM MANAGEMENT	26.	12.	54.	92.
TOTAL	605.	215.	1112.	1933.

ADDITIONAL DATA

DESCRIPTORS

INSTRUCTIONS	40500	APPLICATION	10.952	RESOURCE	3.200
FUNCTIONS	450	STRUCTURE	0.000	LEVEL	0.000

SCHEDULE

COMPLEXITY	1.500				
DESIGN START	JAN 73	IMPL START	0	T&T START	0
DESIGN END	0	IMPL END	0	T&T END	JUN 76

--- PRICE SOFTWARE MODEL ---

DATE 19-OCT-79 TIME 00:36

PROJECT EPA AIR TRAFFIC CONTROL OPERATIONAL

SENSITIVITY DATA

COMPLEXITY

RESOURCE		1.400		1.500		1.600	
3.100	COST	1761.	COST	1840.	COST	1931.
	MONTHS	41.0	MONTHS	41.0	MONTHS	41.0
3.200	COST	1849.	COST	1933.	COST	2030.
	MONTHS	41.0	MONTHS	41.0	MONTHS	41.0
3.300	COST	1938.	COST	2027.	COST	2130.
	MONTHS	41.0	MONTHS	41.0	MONTHS	41.0

SCHEDULE EFFECT SUMMARY

ACTIVITY LENGTH IN MONTHS

COMPLEXITY = 1.500	DESIGN	IMPL	T & I	TOTAL
SPECIFIED SCHEDULE	0.0	0.0	0.0	41.0
(OVERLAP)	(0.0)	(0.0)		
TYPICAL SCHEDULE	12.0	12.9	18.8	29.5
(OVERLAP)	(6.9)	(7.4)		

DEVELOPMENT COSTS

COMPLEXITY = 1.500	DESIGN	IMPL	T & I	TOTAL
SPECIFIED SCHEDULE	605.	215.	1112.	1933.
TYPICAL SCHEDULE	562.	199.	1022.	1784.
ESTIMATED PENALTY	43.	16.	90.	149.

--- PRICE SOFTWARE MODEL ---

DATE 17-OCT-79 TIME 00:39

PROJECT EBB ... AIR TRAFFIC CONTROL UTILITY

INPUT DATA

FILENAME: SIEBB ... DATED: 2 OCT 79

DESCRIPTORS

INSTRUCTIONS	60000	APPLICATION	0.000	RESOURCE	3.200
FUNCTIONS	0	STRUCTURE	0.000	LEVEL	0.000

APPLICATION CATEGORIES		NEW DEVELOPMENT		SYSTEM CONFIGURATION	
	MIX	DESIGN	CODE	TYPES	QUANTITY
DATA S/R	0.00	0.00	0.00	0	0
ONLINE COMM	1.00	0.50	1.00	0	0
REALTIME C&C	0.00	0.00	0.00	0	0
INTERACTIVE	0.00	0.00	0.00	0	0
MATHEMATICAL	0.00	0.00	0.00	***	***
STRING MANIP	0.00	0.00	0.00	***	***
OPR SYSTEMS	0.00	0.00	0.00	***	***

SCHEDULE

COMPLEXITY	1.500				
DESIGN START	JAN 73	IMPL START	0	T&I START	0
DESIGN END	0	IMPL END	0	T&I END	JUN 76

SUPPLEMENTAL INFORMATION

YEAR	1973	ESCALATION	1.000	TECH IMP	1.00
MULTIPLIER	1.300	PLATFORM	1.2	UTILIZATION	0.50

PROGRAM COSTS

COST ELEMENTS	DESIGN	IMPL	T & I	TOTAL
SYSTEMS ENGINEERING	232.	22.	300.	553.
PROGRAMMING	20.	90.	123.	233.
CONFIGURATION CONTROL	72.	29.	181.	281.
DOCUMENTATION	43.	8.	66.	118.
PROGRAM MANAGEMENT	17.	9.	35.	62.
TOTAL	385.	152.	704.	1242.

ADDITIONAL DATA

DESCRIPTORS

INSTRUCTIONS	60000	APPLICATION	6.168	RESOURCE	3.200
FUNCTIONS	662	STRUCTURE	0.000	LEVEL	0.000

SCHEDULE

COMPLEXITY	1.500				
DESIGN START	JAN 73	IMPL START	0	T&I START	0
DESIGN END	0	IMPL END	0	T&I END	JUN 76

--- PRICE SOFTWARE MODEL ---

DATE 19-OCT-79 TIME 00:41

PROJECT FAA AIR TRAFFIC CONTROL UTILITY

SENSITIVITY DATA

COMPLEXITY

RESOURCE		1.400		1.500		1.600	
3.100		COST	1146.	COST	1188.	COST	1237.
		MONTHS	41.0	MONTHS	41.0	MONTHS	41.0
3.200		COST	1202.	COST	1247.	COST	1298.
		MONTHS	41.0	MONTHS	41.0	MONTHS	41.0
3.300		COST	1258.	COST	1306.	COST	1360.
		MONTHS	41.0	MONTHS	41.0	MONTHS	41.0

SCHEDULE EFFECT SUMMARY

ACTIVITY LENGTH IN MONTHS

COMPLEXITY = 1.500	DESIGN		IMPL		T & I	TOTAL
SPECIFIED SCHEDULE	0.0		0.0		0.0	41.0
%OVERLAP		% 0.0%		% 0.0%		
TYPICAL SCHEDULE	11.0		11.8		17.2	26.9
%OVERLAP		% 6.3%		% 6.8%		

DEVELOPMENT COSTS

COMPLEXITY = 1.500	DESIGN	IMPL	T & I	TOTAL
SPECIFIED SCHEDULE	385.	157.	704.	1247.
TYPICAL SCHEDULE	346.	141.	624.	1112.
ESTIMATED PENALTY	39.	16.	80.	135.

--- PRICE SOFTWARE MODEL ---

DATE 14-OCT-79 TIME 00:43

PROJECT EBC AIR TRAFFIC CONTROL DATA REDUCTION

INPUT DATA

FILENAME: STEEL DATED: 2 OCT 79

DESCRIPTORS

INSTRUCTIONS	18000	APPLICATION	0.000	RESOURCE	3.200
FUNCTIONS	0	STRUCTURE	0.000	LEVEL	0.000

APPLICATION CATEGORIES	MIX	NEW DEVELOPMENT DESIGN	CODE	SYSTEM CONFIGURATION TYPES	QUANTITY
DATA S/R	0.00	0.00	0.00	0	0
ONLINE COMM	0.00	0.00	0.00	0	0
REALTIME C&C	0.00	0.00	0.00	0	0
INTERACTIVE	0.00	0.00	0.00	0	0
MATHEMATICAL	0.00	0.00	0.00	***	***
STRING MANIP	1.00	0.50	1.00	***	***
OPR SYSTEMS	0.00	0.00	0.00	***	***

SCHEDULE

COMPLEXITY	1.500				
DESIGN START	JAN 73	IMPL START	0	T&I START	0
DESIGN END	0	IMPL END	0	T&I END	JUN 76

SUPPLEMENTAL INFORMATION

YEAR	1973	ESCALATION	1.000	TECH IMP	1.00
MULTIPLIER	1.300	PLATFORM	1.2	UTILIZATION	0.75

PROGRAM COSTS

COST ELEMENTS	DESIGN	IMPL	T & I	TOTAL
SYSTEMS ENGINEERING	60.	5.	75.	140.
PROGRAMMING	5.	20.	31.	56.
CONFIGURATION CONTROL	16.	6.	40.	62.
DOCUMENTATION	10.	2.	14.	25.
PROGRAM MANAGEMENT	4.	2.	7.	13.
TOTAL	95.	34.	168.	297.

ADDITIONAL DATA

DESCRIPTORS

INSTRUCTIONS	18000	APPLICATION	2.311	RESOURCE	3.200
FUNCTIONS	200	STRUCTURE	0.000	LEVEL	0.000

SCHEDULE

COMPLEXITY	1.500				
DESIGN START	JAN 73	IMPL START	0	T&I START	0
DESIGN END	0	IMPL END	0	T&I END	JUN 76

--- PRICE SOFTWARE MODEL ---

DATE 19-OCT-79 TIME 00:45

PROJECT ERC — AIR TRAFFIC CONTROL DATA REDUCTION

SENSITIVITY DATA

COMPLEXITY

RESOURCE		1.400		1.500		1.600	
3.100		COST	282.	COST	285.	COST	289.
		MONTHS	41.0	MONTHS	41.0	MONTHS	41.0
3.200		COST	294.	COST	297.	COST	301.
		MONTHS	41.0	MONTHS	41.0	MONTHS	41.0
3.300		COST	306.	COST	309.	COST	314.
		MONTHS	41.0	MONTHS	41.0	MONTHS	41.0

SCHEDULE EFFECT SUMMARY

ACTIVITY LENGTH IN MONTHS

COMPLEXITY = 1.500	DESIGN		IMPL		T & I	TOTAL
SPECIFIED SCHEDULE	0.0		0.0		0.0	41.0
(OVERLAP)		(0.0%)		(0.0%)		
TYPICAL SCHEDULE	5.0		5.4		7.8	12.3
(OVERLAP)		(2.9%)		(3.1%)		

DEVELOPMENT COSTS

COMPLEXITY = 1.500	DESIGN	IMPL	T & I	TOTAL
SPECIFIED SCHEDULE	95.	34.	168.	297.
TYPICAL SCHEDULE	60.	21.	102.	182.
ESTIMATED PENALTY	35.	13.	67.	115.

--- PRICE SOFTWARE MODEL ---

DATE 19-OCT-79 TIME 00:48

PROJECT EBD AIR TRAFFIC CONTROL EXERCISE SIMULAT

INPUT DATA

FILENAME: S;EBD DATED: 2 OCT 79

DESCRIPTORS

INSTRUCTIONS	15000	APPLICATION	0.000	RESOURCE	3.200
FUNCTIONS	0	STRUCTURE	0.000	LEVEL	0.000

APPLICATION CATEGORIES		NEW DEVELOPMENT		SYSTEM CONFIGURATION	
	MIX	DESIGN	CODE	TYPES	QUANTITY
DATA S/R	0.00	0.00	0.00	0	0
ONLINE COMM	0.00	0.00	0.00	0	0
REALTIME C&C	1.00	0.50	1.00	0	0
INTERACTIVE	0.00	0.00	0.00	0	0
MATHEMATICAL	0.00	0.00	0.00	***	***
STRING MANIP	0.00	0.00	0.00	***	***
OPR SYSTEMS	0.00	0.00	0.00	***	***

SCHEDULE

COMPLEXITY	1.500				
DESIGN START	JAN 73	IMPL START	0	T&I START	0
DESIGN END	0	IMPL END	0	T&I END	JUN 76

SUPPLEMENTAL INFORMATION

YEAR	1973	ESCALATION	1.000	TECH IMP	1.00
MULTIPLIER	1.300	PLATFORM	1.2	UTILIZATION	0.73

PROGRAM COSTS

COST ELEMENTS	DESIGN	IMPL	T & I	TOTAL
SYSTEMS ENGINEERING	139.	11.	177.	327.
PROGRAMMING	12.	46.	72.	131.
CONFIGURATION CONTROL	32.	13.	92.	142.
DOCUMENTATION	22.	4.	32.	58.
PROGRAM MANAGEMENT	9.	4.	17.	30.
TOTAL	219.	78.	390.	687.

ADDITIONAL DATA

DESCRIPTORS

INSTRUCTIONS	15000	APPLICATION	8.460	RESOURCE	3.200
FUNCTIONS	167	STRUCTURE	0.000	LEVEL	0.000

SCHEDULE

COMPLEXITY	1.500				
DESIGN START	JAN 73	IMPL START	0	T&I START	0
DESIGN END	0	IMPL END	0	T&I END	JUN 76

--- PRICE SOFTWARE MODEL ---

DATE 14-OCT-79 TIME 08:50

PROJECT ERD AIR TRAFFIC CONTROL EXERCISE STANDAT

SENSITIVITY DATA

COMPLEXITY

RESOURCE		1.400	1.500	1.600
3.100	COST	644.	657.	672.
	MONTHS	41.0	41.0	41.0
3.200	COST	673.	687.	703.
	MONTHS	41.0	41.0	41.0
3.300	COST	703.	717.	734.
	MONTHS	41.0	41.0	41.0

SCHEDULE EFFECT SUMMARY

ACTIVITY LENGTH IN MONTHS

COMPLEXITY = 1.500	DESIGN	IMPL	T & I	TOTAL
SPECIFIED SCHEDULE	0.0	0.0	0.0	41.0
%OVERLAP	(0.0%	(0.0%		
TYPICAL SCHEDULE	7.6	8.1	11.9	18.5
%OVERLAP	(4.3%	(4.2%		

DEVELOPMENT COSTS

COMPLEXITY = 1.500	DESIGN	IMPL	T & I	TOTAL
SPECIFIED SCHEDULE	219.	78.	390.	687.
TYPICAL SCHEDULE	160.	56.	277.	493.
ESTIMATED PENALTY	59.	22.	114.	194.

```
                      --- PRICE SOFTWARE MODEL ---
                      DATE 14-OCT-79     TIME 00:53
PROJECT FAA                                    AIR TRAFFIC CONTROL TEST
                              INPUT DATA
FILENAME: STEVE                                          DATED: 2 OCT 79

DESCRIPTORS
  INSTRUCTIONS   18000        APPLICATION   0.000        RESOURCE      3.200
  FUNCTIONS          0        STRUCTURE     0.000        LEVEL         0.000

APPLICATION CATEGORIES         NEW DEVELOPMENT           SYSTEM CONFIGURATION
                   MIX        DESIGN        CODE         TYPES        QUANTITY
  DATA S/R         0.00       0.00          0.00           0               0
  ONLINE COMM      0.00       0.00          0.00           0               0
  REALTIME C&C     1.00       0.50          1.00           0               0
  INTERACTIVE      0.00       0.00          0.00           0               0
  MATHEMATICAL     0.00       0.00          0.00          ***             ***
  STRING MANIP     0.00       0.00          0.00          ***             ***
  OPR SYSTEMS      0.00       0.00          0.00          ***             ***

SCHEDULE
  COMPLEXITY       1.500
  DESIGN START    JAN 73      IMPL START      0          T&I START         0
  DESIGN END         0        IMPL END        0          T&I END      JUN 76

SUPPLEMENTAL INFORMATION
  YEAR             1973       ESCALATION    1.000        TECH IMP      1.00
  MULTIPLIER       1.300      PLATFORM      1.2          UTILIZATION   0.75

                               PROGRAM COSTS

COST ELEMENTS                  DESIGN        IMPL        T & T        TOTAL
  SYSTEMS ENGINEERING            159.         13.         202.         374.
  PROGRAMMING                     14.         53.          83.         149.
  CONFIGURATION CONTROL           43.         15.         108.         166.
  DOCUMENTATION                   26.          4.          38.          68.
  PROGRAM MANAGEMENT              10.          5.          20.          35.
      TOTAL                      252.         89.         450.         792.

                              ADDITIONAL DATA
DESCRIPTORS
  INSTRUCTIONS   18000        APPLICATION   8.460        RESOURCE      3.200
  FUNCTIONS        200        STRUCTURE     0.000        LEVEL         0.000

SCHEDULE
  COMPLEXITY       1.500
  DESIGN START    JAN 73      IMPL START      0          T&I START         0
  DESIGN END         0        IMPL END        0          T&I END      JUN 76
```

--- PRICE SOFTWARE MODEL ---

DATE 19-OCT-79 TIME 00:55

PROJECT EBE AIR TRAFFIC CONTROL TEST

SENSITIVITY DATA

COMPLEXITY

RESOURCE		1.400	1.500	1.600
3.100	COST	740.	757.	776.
	MONTHS	41.0	41.0	41.0
3.200	COST	774.	792.	812.
	MONTHS	41.0	41.0	41.0
3.300	COST	806.	827.	848.
	MONTHS	41.0	41.0	41.0

SCHEDULE EFFECT SUMMARY

ACTIVITY LENGTH IN MONTHS

COMPLEXITY = 1.500	DESIGN		IMPL		T & T	TOTAL
SPECIFIED SCHEDULE	0.0		0.0		0.0	41.0
%OVERLAP		(0.0%)		(0.0%)		
TYPICAL SCHEDULE	8.1		8.7		12.7	19.8
%OVERLAP		(4.6%)		(5.0%)		

DEVELOPMENT COSTS

COMPLEXITY = 1.500	DESIGN	IMPL	T & T	TOTAL
SPECIFIED SCHEDULE	252.	89.	450.	792.
TYPICAL SCHEDULE	191.	67.	334.	592.
ESTIMATED PENALTY	60.	22.	117.	199.

--- PRICE SOFTWARE MODEL ---

DATE 13-OCT-79   TIME 21:52

PROJECT ECA   TACTICAL COMM SYSTEM APPLICATION

INPUT DATA

FILENAME: STECA   DATED: 2 OCT 79

DESCRIPTORS

INSTRUCTIONS	250000	APPLICATION	0.000	RESOURCE	2.900
FUNCTIONS	0	STRUCTURE	0.000	LEVEL	0.000

APPLICATION CATEGORIES		NEW DEVELOPMENT		SYSTEM CONFIGURATION	
	MIX	DESIGN	CODE	TYPES	QUANTITY
DATA S/R	0.00	0.00	0.00	0	0
ONLINE COMM	0.00	0.00	0.00	0	0
REALTIME C&C	1.00	1.00	1.00	0	0
INTERACTIVE	0.00	0.00	0.00	0	0
MATHEMATICAL	0.00	0.00	0.00	***	***
STRING MANIP	0.00	0.00	0.00	***	***
OPR SYSTEMS	0.00	0.00	0.00	***	***

SCHEDULE

COMPLEXITY	1.300				
DESIGN START	APR 74	IMPL START	0	T&T START	0
DESIGN END	0	IMPL END	0	T&T END	FEB 77

SUPPLEMENTAL INFORMATION

YEAR	1974	ESCALATION	1.000	TECH IMP	1.00
MULTIPLIER	1.300	PLATFORM	1.4	UTILIZATION	0.50

PROGRAM COSTS

COST ELEMENTS	DESIGN	IMPL	T & T	TOTAL
SYSTEMS ENGINEERING	2756.	206.	2668.	5630.
PROGRAMMING	331.	949.	1091.	2371.
CONFIGURATION CONTROL	812.	396.	1925.	3133.
DOCUMENTATION	600.	111.	835.	1545.
PROGRAM MANAGEMENT	314.	114.	425.	854.
TOTAL	4813.	1776.	6995.	13534.

ADDITIONAL DATA

DESCRIPTORS

INSTRUCTIONS	250000	APPLICATION	8.460	RESOURCE	2.900
FUNCTIONS	2778	STRUCTURE	0.000	LEVEL	0.000

SCHEDULE

COMPLEXITY	1.300				
DESIGN START	APR 74	IMPL START	0	T&T START	0
DESIGN END	0	IMPL END	0	T&T END	FEB 77

--- PRICE SOFTWARE MODEL ---

DATE 13-OCT-79 TIME 21:53

PROJECT ECA TACTICAL COMM SYSTEM APPLICATION

SENSITIVITY DATA

COMPLEXITY

RESOURCE		1.200	1.300	1.400
2.800	COST	10395.	12649.	15304.
	MONTHS	34.0	34.0	34.0
2.900	COST	11114.	13534.	16351.
	MONTHS	34.0	34.0	34.0
3.000	COST	11856.	14442.	17493.
	MONTHS	34.0	34.0	34.0

SCHEDULE EFFECT SUMMARY

ACTIVITY LENGTH IN MONTHS

COMPLEXITY = 1.300	DESIGN		IMPL		T & T	TOTAL
SPECIFIED SCHEDULE	0.0		0.0		0.0	34.0
(OVERLAP)		(0.0%)		(0.0%)		
TYPICAL SCHEDULE	24.0		24.2		34.3	54.3
(OVERLAP)		(14.2%)		(14.6%)		

DEVELOPMENT COSTS

COMPLEXITY = 1.300	DESIGN	IMPL	T & T	TOTAL
SPECIFIED SCHEDULE	4813.	1726.	6995.	13534.
TYPICAL SCHEDULE	3479.	1064.	5327.	9870.
ESTIMATED PENALTY	1335.	661.	1669.	3665.

--- PRICE SOFTWARE MODEL ---

DATE 13-OCT-79 TIME 21:59

PROJECT ECP — TACTICAL COMM SYSTEM SUPPORT

INPUT DATA

FILENAME: STECP — DATED: 2 OCT 79

DESCRIPTORS

INSTRUCTIONS	30000	APPLICATION	0.000	RESOURCE	2.900
FUNCTIONS	0	STRUCTURE	0.000	LEVEL	0.000

APPLICATION CATEGORIES		NEW DEVELOPMENT		SYSTEM CONFIGURATION	
	MIX	DESIGN	CODE	TYPES	QUANTITY
DATA S/R	0.00	0.00	0.00	0	0
ONLINE COMM	1.00	1.00	1.00	0	0
REALTIME C&C	0.00	0.00	0.00	0	0
INTERACTIVE	0.00	0.00	0.00	0	0
MATHEMATICAL	0.00	0.00	0.00	***	***
STRING MANIP	0.00	0.00	0.00	***	***
OPR SYSTEMS	0.00	0.00	0.00	***	***

SCHEDULE

COMPLEXITY	1.300				
DESIGN START	APR 74	IMPL START	0	T&T START	0
DESIGN END	0	IMPL END	0	T&T END	FEB 77

SUPPLEMENTAL INFORMATION

YEAR	1974	ESCALATION	1.000	TECH IMP	1.00
MULTIPLIER	1.300	PLATFORM	1.4	UTILIZATION	0.50

PROGRAM COSTS

COST ELEMENTS	DESIGN	IMPL	T & I	TOTAL
SYSTEMS ENGINEERING	200.	12.	182.	394.
PROGRAMMING	24.	56.	75.	154.
CONFIGURATION CONTROL	48.	17.	111.	176.
DOCUMENTATION	34.	5.	44.	83.
PROGRAM MANAGEMENT	18.	5.	22.	45.
TOTAL	325.	95.	433.	852.

ADDITIONAL DATA

DESCRIPTORS

INSTRUCTIONS	30000	APPLICATION	6.168	RESOURCE	2.900
FUNCTIONS	333	STRUCTURE	0.000	LEVEL	0.000

SCHEDULE

COMPLEXITY	1.300				
DESIGN START	APR 74	IMPL START	0	T&T START	0
DESIGN END	0	IMPL END	0	T&T END	FEB 77

--- PRICE SOFTWARE MODEL ---

DATE 13-OCT-79 TIME 22:00

PROJECT ECB TACTICAL COMM SYSTEM SUPPORT

SENSITIVITY DATA

COMPLEXITY

RESOURCE		1.200		1.300		1.400	
2.800	COST	782.	COST	811.	COST	845.
	MONTHS	34.0	MONTHS	34.0	MONTHS	34.0
2.900	COST	822.	COST	852.	COST	889.
	MONTHS	34.0	MONTHS	34.0	MONTHS	34.0
3.000	COST	862.	COST	895.	COST	934.
	MONTHS	34.0	MONTHS	34.0	MONTHS	34.0

SCHEDULE EFFECT SUMMARY

ACTIVITY LENGTH IN MONTHS

COMPLEXITY = 1.300	DESIGN	IMPL	T & I	TOTAL
SPECIFIED SCHEDULE	0.0	0.0	0.0	34.0
(OVERLAP)	(0.0%)	(0.0%)		
TYPICAL SCHEDULE	9.2	10.0	13.9	22.0
(OVERLAP)	(5.8%)	(5.9%)		

DEVELOPMENT COSTS

COMPLEXITY = 1.300	DESIGN	IMPL	T & I	TOTAL
SPECIFIED SCHEDULE	325.	95.	433.	852.
TYPICAL SCHEDULE	291.	84.	383.	759.
ESTIMATED PENALTY	33.	10.	50.	93.

--- PRICE SOFTWARE MODEL ---

DATE 14-OCT-79 TIME 15:33

PROJECT CB ATTACK WARNING SYSTEM

PROGRAM COSTS

COST ELEMENTS	DESIGN	TMRL	T & T	TOTAL
SYSTEMS ENGINEERING	14605.	865.	17356.	32826.
PROGRAMMING	2548.	4489.	7099.	14136.
CONFIGURATION CONTROL	3661.	1720.	14651.	20032.
DOCUMENTATION	2932.	528.	6160.	9619.
PROGRAM MANAGEMENT	2333.	568.	3343.	6244.
TOTAL	26079.	8169.	48614.	82862.

ADDITIONAL DATA

DESCRIPTORS

INSTRUCTIONS	2250000	APPLICATION	8.460	RESOURCE	3.000
FUNCTIONS	30556	STRUCTURE	0.000	LEVEL	0.000

SCHEDULE

COMPLEXITY	1.100				
DESIGN START	JAN 72	TMRL START	0	T&T START	0
DESIGN END	0	TMRL END	0	T&T END	DEC 79

SENSITIVITY DATA

COMPLEXITY

RESOURCE		1.000	1.100	1.200
2.900	COST	68625.	77417.	88969.
	MONTHS	95.0	95.0	95.0
3.000	COST	73404.	82862.	95495.
	MONTHS	95.0	95.0	95.0
3.100	COST	77600.	82656.	101288.
	MONTHS	95.0	95.0	95.0

--- PRICE SOFTWARE MODEL ---

DATE 19-OCT-79 TIME 15:34

PROJECT CA ATTACK WARNING SYSTEM

SCHEDULE EFFECT SUMMARY

ACTIVITY LENGTH IN MONTHS

COMPLEXITY = 1.100	DESIGN	IMPL	T & T	TOTAL
SPECIFIED SCHEDULE	0.0	0.0	0.0	95.0
%OVERLAP	% 0.0%	% 0.0%		
TYPICAL SCHEDULE	43.1	42.0	58.6	100.5
%OVERLAP	%21.9%	%21.4%		

DEVELOPMENT COSTS

COMPLEXITY = 1.100	DESIGN	IMPL	T & T	TOTAL
SPECIFIED SCHEDULE	26079.	8169.	48614.	82862.
TYPICAL SCHEDULE	26040.	8203.	49614.	83856.
ESTIMATED PENALTY	39.	-33.	-1000.	-994.

--- PRICE SOFTWARE MODEL ---

DATE 14-OCT-79 TIME 15:36

PROJECT CB GROUND BASED RADAR

PROGRAM COSTS

COST ELEMENTS	DESIGN	IMPL	T & I	TOTAL
SYSTEMS ENGINEERING	249.	13.	222.	484.
PROGRAMMING	30.	61.	91.	182.
CONFIGURATION CONTROL	55.	17.	122.	194.
DOCUMENTATION	35.	5.	44.	84.
PROGRAM MANAGEMENT	19.	5.	23.	47.
TOTAL	388.	101.	501.	991.

ADDITIONAL DATA

DESCRIPTORS

INSTRUCTIONS	25000	APPLICATION	8.460	RESOURCE	3.000
FUNCTIONS	.278	STRUCTURE	0.000	LEVEL	0.000

SCHEDULE

COMPLEXITY	1.300				
DESIGN START	JAN 75	IMPL START	0	T&I START	0
DESIGN END	0	IMPL END	0	T&I END	JUN 76

SENSITIVITY DATA

COMPLEXITY

RESOURCE		1.200	1.300	1.400
2.900	COST	834.	932.	1024.
	MONTHS	17.0	17.0	17.0
3.000	COST	880.	991.	1138.
	MONTHS	17.0	17.0	17.0
3.100	COST	922.	1046.	1203.
	MONTHS	17.0	17.0	17.0

--- PRICE SOFTWARE MODEL ---

DATE 17-OCT-79 TIME 15:38

PROJECT CE GROUND BASED RADAR

SCHEDULE EFFECT SUMMARY

ACTIVITY LENGTH IN MONTHS

COMPLEXITY = 1.300	DESIGN		IMPL		T & I	TOTAL
SPECIFIED SCHEDULE	0.0		0.0		0.0	17.0
%OVERLAP		(0.0%)		(0.0%)		
TYPICAL SCHEDULE	8.8		8.5		11.8	19.8
%OVERLAP		(4.7%)		(4.5%)		

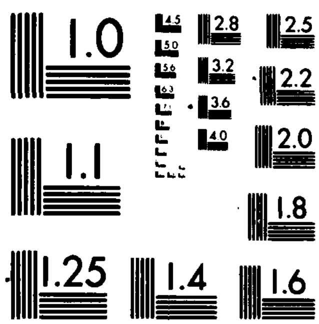
DEVELOPMENT COSTS

COMPLEXITY = 1.300	DESIGN	IMPL	T & I	TOTAL
SPECIFIED SCHEDULE	388.	101.	501.	991.
TYPICAL SCHEDULE	372.	94.	483.	949.
ESTIMATED PENALTY	16.	7.	18.	42.

AD-A083 713 AIR FORCE INST OF TECH WRIGHT-PATTERSON AFB OH SCHOO--ETC F/G 14/1
AN ANALYSIS OF THE RCA PRICE-S COST ESTIMATION MODEL AS IT RELA--ETC(U)
DEC 79 R E STEFFEY
UNCLASSIFIED AFIT/GSM/SM/79D-20 NL

3 of 3

END
DATE
FILMED
6-80
DTIC



MICROCOPY RESOLUTION TEST CHART

NATIONAL BUREAU [illegible]

--- PRICE SOFTWARE MODEL ---

DATE 14-OCT-79 TIME 15:40

PROJECT CC INTELLIGENCE PROCESSING SYSTEM

PROGRAM COSTS

COST ELEMENTS	DESIGN	IMPL	T & T	TOTAL
SYSTEMS ENGINEERING	1786.	121.	1697.	3604.
PROGRAMMING	256.	591.	694.	1542.
CONFIGURATION CONTROL	423.	192.	1163.	1778.
DOCUMENTATION	300.	55.	443.	799.
PROGRAM MANAGEMENT	198.	61.	240.	498.
TOTAL	2963.	1020.	4236.	8220.

ADDITIONAL DATA

DESCRIPTORS

INSTRUCTIONS	250000	APPLICATION	8.460	RESOURCE	2.900
FUNCTIONS	2778	STRUCTURE	0.000	LEVEL	0.000

SCHEDULE

COMPLEXITY	1.200				
DESIGN START	JAN 75	IMPL START	0	T&T START	0
DESIGN END	0	IMPL END	0	T&T END	JUN 77

SENSITIVITY DATA

COMPLEXITY

RESOURCE		1.100	1.200	1.300
2.800	COST	6391.	7681.	9328.
	MONTHS	29.0	29.0	29.0
2.900	COST	6776.	8220.	9989.
	MONTHS	29.0	29.0	29.0
3.000	COST	7224.	8776.	10673.
	MONTHS	29.0	29.0	29.0

--- PRICE SOFTWARE MODEL ---

DATE 14-OCT-79 TIME 15:41

PROJECT CC INTELLIGENCE PROCESSING SYSTEM

SCHEDULE EFFECT SUMMARY

ACTIVITY LENGTH IN MONTHS

COMPLEXITY = 1.200	DESIGN	IMPL	T & I	TOTAL
SPECIFIED SCHEDULE	0.0	0.0	0.0	29.0
(OVERLAP)	(0.0%)	(0.0%)		
TYPICAL SCHEDULE	18.0	17.3	23.9	41.3
(OVERLAP)	(9.2%)	(6.8%)		

DEVELOPMENT COSTS

COMPLEXITY = 1.200	DESIGN	IMPL	T & I	TOTAL
SPECIFIED SCHEDULE	2963.	1020.	4236.	8220.
TYPICAL SCHEDULE	2420.	725.	3533.	6678.
ESTIMATED PENALTY	543.	295.	704.	1542.

--- PRICE SOFTWARE MODEL ---

DATE 17-OCT-79 TIME 15:43

PROJECT CD SEA SURVEILLANCE SYSTEM

PROGRAM COSTS

COST ELEMENTS	DESIGN	IMPL	T & I	TOTAL
SYSTEMS ENGINEERING	1235.	71.	1139.	2445.
PROGRAMMING	267.	392.	466.	1125.
CONFIGURATION CONTROL	218.	120.	802.	1140.
DOCUMENTATION	194.	37.	333.	563.
PROGRAM MANAGEMENT	177.	37.	169.	382.
TOTAL	2090.	656.	2908.	5655.

ADDITIONAL DATA

DESCRIPTORS

INSTRUCTIONS	140000	APPLICATION	8.460	RESOURCE	3.100
FUNCTIONS	1556	STRUCTURE	0.000	LEVEL	0.000

SCHEDULE

COMPLEXITY	1.000				
DESIGN START	JAN 75	IMPL START	0	T&I START	0
DESIGN END	0	IMPL END	0	T&I END	DEC 76

SENSITIVITY DATA

COMPLEXITY

RESOURCE		.900		1.000		1.100	
3.000	COST	4301.	COST	5326.	COST	6632.
	MONTHS	23.0	MONTHS	23.0	MONTHS	23.0
3.100	COST	4559.	COST	5655.	COST	7052.
	MONTHS	23.0	MONTHS	23.0	MONTHS	23.0
3.200	COST	4823.	COST	5992.	COST	7480.
	MONTHS	23.0	MONTHS	23.0	MONTHS	23.0

--- PRICE SOFTWARE MODEL ---

DATE 14-OCT-79 TIME 15:45

PROJECT CD SEA SURVEILLANCE SYSTEM

SCHEDULE EFFECT SUMMARY

ACTIVITY LENGTH IN MONTHS

COMPLEXITY = 1.000	DESIGN		IMPL		I & T	TOTAL
SPECIFIED SCHEDULE	0.0		0.0		0.0	23.0
(OVERLAP)		(0.0%)		(0.0%)		
TYPICAL SCHEDULE	13.8		13.9		19.6	32.6
(OVERLAP)		(7.3%)		(7.4%)		

DEVELOPMENT COSTS

COMPLEXITY = 1.000	DESIGN	IMPL	I & T	TOTAL
SPECIFIED SCHEDULE	2090.	656.	2908.	5655.
TYPICAL SCHEDULE	1708.	473.	2420.	4601.
ESTIMATED PENALTY	382.	183.	488.	1053.

--- PRICE SOFTWARE MODEL ---

DATE 17-OCT-79 TIME 16:56

PROJECT CE MISSILE C AND C SYSTEM

PROGRAM COSTS

COST ELEMENTS	DESIGN	IMPL	T & I	TOTAL
SYSTEMS ENGINEERING	151.	8.	134.	293.
PROGRAMMING	18.	36.	55.	109.
CONFIGURATION CONTROL	33.	10.	73.	115.
DOCUMENTATION	23.	3.	28.	54.
PROGRAM MANAGEMENT	12.	3.	14.	28.
TOTAL	237.	59.	304.	599.

ADDITIONAL DATA

DESCRIPTORS

INSTRUCTIONS	11000	APPLICATION	8.460	RESOURCE	3.100
FUNCTIONS	122	STRUCTURE	0.000	LEVEL	0.000

SCHEDULE

COMPLEXITY	1.300				
DESIGN START	JAN 75	IMPL START	0	T&I START	0
DESIGN END	0	IMPL END	0	T&I END	DEC 78

SENSITIVITY DATA

COMPLEXITY

RESOURCE		1.200	1.300	1.400
3.000	COST	547.	573.	603.
	MONTHS	23.0	23.0	23.0
3.100	COST	572.	599.	632.
	MONTHS	23.0	23.0	23.0
3.200	COST	598.	627.	661.
	MONTHS	23.0	23.0	23.0

--- PRICE SOFTWARE MODEL ---

DATE 19-OCT-79 TIME 16:58

PROJECT CE MISSILE C AND C SYSTEM

SCHEDULE EFFECT SUMMARY

ACTIVITY LENGTH IN MONTHS

COMPLEXITY = 1.300	DESIGN	IMPL	T & I	TOTAL
SPECIFIED SCHEDULE	0.0	0.0	0.0	23.0
%OVERLAP%		(0.0%)	(0.0%)	
TYPICAL SCHEDULE	7.7	8.0	11.1	17.5
%OVERLAP%		(4.6%)	(4.8%)	

DEVELOPMENT COSTS

COMPLEXITY = 1.300	DESIGN	IMPL	T & I	TOTAL
SPECIFIED SCHEDULE	237.	59.	304.	599.
TYPICAL SCHEDULE	225.	56.	289.	570.
ESTIMATED PENALTY	11.	3.	15.	29.

--- PRICE SOFTWARE MODEL ---

DATE 19-OCT-79 TIME 15:50

PROJECT CE STRATEGIC COMMUNICATIONS SYSTEM

PROGRAM COSTS

COST ELEMENTS	DESIGN	INTEG	T & T	TOTAL
SYSTEMS ENGINEERING	1356.	80.	1290.	2726.
PROGRAMMING	195.	393.	528.	1115.
CONFIGURATION CONTROL	317.	128.	873.	1315.
DOCUMENTATION	224.	36.	331.	592.
PROGRAM MANAGEMENT	148.	40.	179.	366.
TOTAL	2239.	675.	3201.	6114.

ADDITIONAL DATA

DESCRIPTORS

INSTRUCTIONS	214000	APPLICATION	8.460	RESOURCE	3.000
FUNCTIONS	2328	STRUCTURE	0.000	LEVEL	0.000

SCHEDULE

COMPLEXITY	1.200				
DESIGN START	JAN 75	INTEG START	0	T&T START	0
DESIGN END	0	INTEG END	0	T&T END	DEC 77

SENSITIVITY DATA

COMPLEXITY

RESOURCE	1.100	1.200	1.300
2.900	COST 5144. MONTHS 35.0	COST 5752. MONTHS 35.0	COST 6624. MONTHS 35.0
3.000	COST 5453. MONTHS 35.0	COST 6114. MONTHS 35.0	COST 7056. MONTHS 35.0
3.100	COST 5770. MONTHS 35.0	COST 6487. MONTHS 35.0	COST 7501. MONTHS 35.0

--- PRICE SOFTWARE MODEL ---

DATE 14-OCT-79 TIME 15:52

PROJECT CE STRATEGIC COMMUNICATIONS SYSTEM

SCHEDULE EFFECT SUMMARY

ACTIVITY LENGTH IN MONTHS

COMPLEXITY = 1.200	DESIGN	IMPL	T & I	TOTAL
SPECIFIED SCHEDULE	0.0	0.0	0.0	35.0
%OVERLAP	(0.0%)	(0.0%)		
TYPICAL SCHEDULE	17.1	16.5	22.8	39.2
%OVERLAP	(8.2%)	(8.4%)		

DEVELOPMENT COSTS

COMPLEXITY = 1.200	DESIGN	IMPL	T & I	TOTAL
SPECIFIED SCHEDULE	2239.	675.	3201.	6114.
TYPICAL SCHEDULE	2195.	651.	3172.	6018.
ESTIMATED PENALTY	45.	24.	29.	97.

--- PRICE SOFTWARE MODEL ---

ECIRP (CALIBRATION) MODE

DATE 16-OCT-79 TIME 23:22

PROJECT 838 INERTIAL NAVIGATION SYSTEM

PROGRAM COSTS

COST ELEMENTS	DESIGN	IMPL	T & I	TOTAL
SYSTEMS ENGINEERING	97.	5.	89.	191.
PROGRAMMING	21.	28.	36.	85.
CONFIGURATION CONTROL	16.	8.	56.	80.
DOCUMENTATION	15.	3.	25.	42.
PROGRAM MANAGEMENT	12.	2.	11.	26.
TOTAL	161.	46.	218.	425.

ADDITIONAL DATA

DESCRIPTORS					
INSTRUCTIONS	16000	APPLICATION	6.461	RESOURCE	2.787
FUNCTIONS	200	STRUCTURE	0.000	LEVEL	0.000

SCHEDULE					
COMPLEXITY	1.000				
DESIGN START	MAY 73	IMPL START	0	T&I START	0
DESIGN END	0	IMPL END	0	T&I END	DEC 75

--- PRICE SOFTWARE MODEL ---

DATE 16-OCT-79 TIME 23:39

PROJECT ASA — INERTIAL NAVIGATION SYSTEM

INPUT DATA

FILENAME: STASA — DATED: 16 OCT 79

DESCRIPTORS

INSTRUCTIONS	18000	APPLICATION	0.000	RESOURCE	2.800
FUNCTIONS	0	STRUCTURE	0.000	LEVEL	0.000

APPLICATION CATEGORIES		NEW DEVELOPMENT		SYSTEM CONFIGURATION	
	MIX	DESIGN	CODE	TYPES	QUANTITY
DATA S/R	0.07	0.00	1.00	0	0
ONLINE COMM	0.00	0.00	0.00	0	0
REALTIME C&C	0.17	1.00	1.00	0	0
INTERACTIVE	0.07	1.00	1.00	0	0
MATHEMATICAL	0.33	0.60	1.00	***	***
STRING MANIP	0.03	0.00	1.00	***	***
OPR SYSTEMS	0.33	1.00	1.00	***	***

SCHEDULE

COMPLEXITY	1.000				
DESIGN START	MAY 73	IMPL START	0	T&T START	0
DESIGN END	0	IMPL END	0	T&T END	DEC 75

SUPPLEMENTAL INFORMATION

YEAR	1973	ESCALATION	1.000	TECH IMP	0.00
MULTIPLIER	1.000	PLATFORM	1.2	UTILIZATION	0.50

PROGRAM COSTS

COST ELEMENTS	DESIGN	IMPL	T & T	TOTAL
SYSTEMS ENGINEERING	98.	5.	89.	193.
PROGRAMMING	21.	28.	37.	86.
CONFIGURATION CONTROL	16.	8.	57.	80.
DOCUMENTATION	15.	3.	25.	43.
PROGRAM MANAGEMENT	12.	2.	12.	26.
TOTAL	162.	46.	219.	428.

ADDITIONAL DATA

DESCRIPTORS

INSTRUCTIONS	18000	APPLICATION	6.361	RESOURCE	2.800
FUNCTIONS	200	STRUCTURE	0.000	LEVEL	0.000

SCHEDULE

COMPLEXITY	1.000				
DESIGN START	MAY 73	IMPL START	0	T&T START	0
DESIGN END	0	IMPL END	0	T&T END	DEC 75

--- PRICE SOFTWARE MODEL ---

DATE 16-OCT-79 TIME 23:40

PROJECT ASA INERTIAL NAVIGATION SYSTEM

SENSITIVITY DATA

COMPLEXITY

RESOURCE		.900		1.000		1.100	
2.700		COST	398.	COST	406.	COST	419.
		MONTHS	31.0	MONTHS	31.0	MONTHS	31.0
2.800		COST	419.	COST	428.	COST	441.
		MONTHS	31.0	MONTHS	31.0	MONTHS	31.0
2.900		COST	440.	COST	450.	COST	464.
		MONTHS	31.0	MONTHS	31.0	MONTHS	31.0

SCHEDULE EFFECT SUMMARY

ACTIVITY LENGTH IN MONTHS

COMPLEXITY = 1.000	DESIGN	IMPL	T & I	TOTAL
SPECIFIED SCHEDULE	0.0	0.0	0.0	31.0
(OVERLAP)	(0.0%)	(0.0%)		
TYPICAL SCHEDULE	7.1	7.8	11.2	17.1
(OVERLAP)	(4.3%)	(4.7%)		

DEVELOPMENT COSTS

COMPLEXITY = 1.000	DESIGN	IMPL	T & I	TOTAL
SPECIFIED SCHEDULE	162.	46.	219.	428.
TYPICAL SCHEDULE	135.	38.	180.	354.
ESTIMATED PENALTY	27.	8.	39.	74.

--- PRICE SOFTWARE MODEL ---

DATE 16-OCT-79 TIME 23:44

PROJECT ASB NAVIGATION SYSTEM SIMULATION

INPUT DATA

FILENAME: STASB DATED: 16 OCT 79

DESCRIPTORS

INSTRUCTIONS	2600	APPLICATION	0.000	RESOURCE	2.900
FUNCTIONS	0	STRUCTURE	0.000	LEVEL	0.000

APPLICATION CATEGORIES		NEW DEVELOPMENT		SYSTEM CONFIGURATION	
	MIX	DESIGN	CODE	TYPES	QUANTITY
DATA S/R	0.00	0.00	0.00	0	0
ONLINE COMM	0.00	0.00	0.00	0	0
REALTIME C&C	0.00	0.00	0.00	0	0
INTERACTIVE	0.05	1.00	1.00	0	0
MATHEMATICAL	0.95	0.10	1.00	***	***
STRING MANIP	0.00	0.00	0.00	***	***
OPR SYSTEMS	0.00	0.00	0.00	***	***

SCHEDULE

COMPLEXITY	0.900				
DESIGN START	NOV 77	IMPL START	0	T&I START	0
DESIGN END	0	IMPL END	0	T&I END	FEB 78

SUPPLEMENTAL INFORMATION

YEAR	1977	ESCALATION	1.000	TECH IMP	0.00
MULTIPLIER	1.000	PLATFORM	1.0	UTILIZATION	0.30

PROGRAM COSTS

COST ELEMENTS	DESIGN	IMPL	T & I	TOTAL
SYSTEMS ENGINEERING	2.	0.	3.	6.
PROGRAMMING	1.	1.	1.	3.
CONFIGURATION CONTROL	0.	0.	1.	2.
DOCUMENTATION	0.	0.	0.	1.
PROGRAM MANAGEMENT	0.	0.	0.	0.
TOTAL	3.	1.	6.	11.

ADDITIONAL DATA

DESCRIPTORS

INSTRUCTIONS	2600	APPLICATION	1.370	RESOURCE	2.900
FUNCTIONS	29	STRUCTURE	0.000	LEVEL	0.000

SCHEDULE

COMPLEXITY	0.900				
DESIGN START	NOV 77	IMPL START	0	T&I START	0
DESIGN END	0	IMPL END	0	T&I END	FEB 78

--- PRICE SOFTWARE MODEL ---

DATE 16-OCT-79 TIME 23:46

PROJECT ASB NAVIGATION SYSTEM SIMULATION

SENSITIVITY DATA

COMPLEXITY

RESOURCE	.800		.900		1.000	
2.800	COST	11.	COST	11.	COST	11.
	MONTHS	3.0	MONTHS	3.0	MONTHS	3.0
2.900	COST	11.	COST	11.	COST	11.
	MONTHS	3.0	MONTHS	3.0	MONTHS	3.0
3.000	COST	11.	COST	11.	COST	12.
	MONTHS	3.0	MONTHS	3.0	MONTHS	3.0

SCHEDULE EFFECT SUMMARY

ACTIVITY LENGTH IN MONTHS

COMPLEXITY = 0.900	DESIGN	IMPL	T & I	TOTAL
SPECIFIED SCHEDULE	0.0	0.0	0.0	3.0
%OVERLAP	% 0.0%	% 0.0%		
TYPICAL SCHEDULE	0.8	0.2	1.1	2.1
%OVERLAP	% 0.2%	% 0.2%		

DEVELOPMENT COSTS

COMPLEXITY = 0.900	DESIGN	IMPL	T & I	TOTAL
SPECIFIED SCHEDULE	3.	1.	6.	11.
TYPICAL SCHEDULE	3.	1.	6.	11.
ESTIMATED PENALTY	0.	0.	0.	0.

--- PRICE SOFTWARE MODEL ---

DATE 16-OCT-79   TIME 23:50

PROJECT ASC — AIRBORNE RADAR SYSTEM

INPUT DATA

FILENAME: STASC — DATED: 16 OCT 79

DESCRIPTORS

INSTRUCTIONS	23250	APPLICATION	0.000	RESOURCE	2.200
FUNCTIONS	0	STRUCTURE	0.000	LEVEL	0.000

APPLICATION CATEGORIES		NEW DEVELOPMENT		SYSTEM CONFIGURATION	
	MIX	DESIGN	CODE	TYPES	QUANTITY
DATA S/R	0.15	1.00	1.00	0	0
ONLINE COMM	0.00	0.00	0.00	0	0
REALTIME C&C	0.15	1.00	1.00	0	0
INTERACTIVE	0.05	1.00	1.00	0	0
MATHEMATICAL	0.65	0.99	0.99	***	***
STRING MANIP	0.00	0.00	0.00	***	***
OPR SYSTEMS	0.00	0.00	0.00	***	***

SCHEDULE

COMPLEXITY	0.900				
DESIGN START	MAY 74	IMPL START	0	I&T START	0
DESIGN END	0	IMPL END	0	I&T END	MAY 79

SUPPLEMENTAL INFORMATION

YEAR	1974	ESCALATION	1.000	TECH IMP	0.00
MULTIPLIER	1.120	PLATFORM	1.2	UTILIZATION	0.85

PROGRAM COSTS

COST ELEMENTS	DESIGN	IMPL	I & T	TOTAL
SYSTEMS ENGINEERING	452.	16.	432.	901.
PROGRAMMING	124.	98.	177.	398.
CONFIGURATION CONTROL	69.	30.	316.	415.
DOCUMENTATION	75.	10.	147.	231.
PROGRAM MANAGEMENT	79.	9.	69.	156.
TOTAL	799.	163.	1141.	2102.

ADDITIONAL DATA

DESCRIPTORS

INSTRUCTIONS	23250	APPLICATION	2.995	RESOURCE	2.200
FUNCTIONS	819	STRUCTURE	0.000	LEVEL	0.000

SCHEDULE

COMPLEXITY	0.900				
DESIGN START	MAY 74	IMPL START	0	I&T START	0
DESIGN END	0	IMPL END	0	I&T END	MAY 79

--- PRICE SOFTWARE MODEL ---

DATE 16-OCT-79 TIME 23:52

PROJECT ASC AIRBORNE RADAR SYSTEM

SENSITIVITY DATA

COMPLEXITY

RESOURCE		.800	.900	1.000
2.600	COST	2007.	2003.	2021.
	MONTHS	60.0	60.0	60.0
2.700	COST	2105.	2102.	2121.
	MONTHS	60.0	60.0	60.0
2.800	COST	2221.	2218.	2238.
	MONTHS	60.0	60.0	60.0

SCHEDULE EFFECT SUMMARY

ACTIVITY LENGTH IN MONTHS

COMPLEXITY = 0.900	DESIGN	IMPL	T & I	TOTAL
SPECIFIED SCHEDULE	0.0	0.0	0.0	60.0
%OVERLAP	(0.0%)	(0.0%)		
TYPICAL SCHEDULE	8.2	9.3	13.4	21.1
%OVERLAP	(5.0%)	(5.3%)		

DEVELOPMENT COSTS

COMPLEXITY = 0.900	DESIGN	IMPL	T & I	TOTAL
SPECIFIED SCHEDULE	799.	163.	1141.	2102.
TYPICAL SCHEDULE	524.	104.	703.	1331.
ESTIMATED PENALTY	275.	59.	437.	771.

--- PRICE SOFTWARE MODEL ---

ECIRP (CALIBRATION) MODE

DATE 16-OCT-79 TIME 18:46

PROJECT DCA OPERATIONAL PLANNING SYSTEM

PROGRAM COSTS

COST ELEMENTS	DESIGN	IMPL	T & I	TOTAL
SYSTEMS ENGINEERING	97.	6.	90.	192.
PROGRAMMING	21.	31.	37.	89.
CONFIGURATION CONTROL	14.	8.	53.	75.
DOCUMENTATION	10.	2.	16.	27.
PROGRAM MANAGEMENT	11.	2.	10.	23.
TOTAL	152.	49.	205.	406.

ADDITIONAL DATA

DESCRIPTORS

INSTRUCTIONS	315000	APPLICATION	1.588	RESOURCE	1.476
FUNCTIONS	3500	STRUCTURE	0.000	LEVEL	0.000

SCHEDULE

COMPLEXITY	1.000				
DESIGN START	FEB 78	IMPL START	0	T&I START	0
DESIGN END	0	IMPL END	0	T&I END	MAR 82

--- PRICE SOFTWARE MODEL ---

ECIRP (CALIBRATION) MODE

DATE 16-OCT-79 TIME 20:52

PROJECT DCA ENGINEERING MANAGEMENT SYSTEM

PROGRAM COSTS

COST ELEMENTS	DESIGN	IMPL	T & I	TOTAL
SYSTEMS ENGINEERING	16.	1.	14.	31.
PROGRAMMING	3.	5.	6.	14.
CONFIGURATION CONTROL	2.	1.	7.	10.
DOCUMENTATION	1.	0.	2.	4.
PROGRAM MANAGEMENT	1.	0.	1.	3.
TOTAL	24.	8.	30.	62.

ADDITIONAL DATA

DESCRIPTORS

INSTRUCTIONS	66000	APPLICATION	1.558	RESOURCE	1.282
FUNCTIONS	233	STRUCTURE	0.000	LEVEL	0.000

SCHEDULE

COMPLEXITY	1.000				
DESIGN START	JAN 75	IMPL START	0	T&I START	0
DESIGN END	0	IMPL END	0	T&I END	DEC 76

--- PRICE SOFTWARE MODEL ---

ECIRP (CALIBRATION) MODE

DATE 12-OCT-79 TIME 22:29

PROJECT ACB MODIFICATION MANAGEMENT SYSTEM

PROGRAM COSTS

COST ELEMENTS	DESIGN	IMPL	T & I	TOTAL
SYSTEMS ENGINEERING	9.	0.	2.	15.
PROGRAMMING	2.	3.	3.	7.
CONFIGURATION CONTROL	1.	1.	3.	5.
DOCUMENTATION	1.	0.	1.	2.
PROGRAM MANAGEMENT	1.	0.	1.	1.
TOTAL	13.	4.	15.	31.

ADDITIONAL DATA

DESCRIPTORS

INSTRUCTIONS	27900	APPLICATION	1.588	RESOURCE	1.352
FUNCTIONS	310	STRUCTURE	0.000	LEVEL	0.000

SCHEDULE

COMPLEXITY	1.000				
DESIGN START	MAY 78	IMPL START	0	T&I START	0
DESIGN END	0	IMPL END	0	T&I END	JAN 79

--- PRICE SOFTWARE MODEL ---

ECIRP (CALIBRATION) MODE

DATE 13-OCT-79 TIME 23:29

PROJECT ACB MATERIAL REPORTING SYSTEM

PROGRAM COSTS

COST ELEMENTS	DESIGN	IMPL	T & I	TOTAL
SYSTEMS ENGINEERING	31.	1.	27.	59.
PROGRAMMING	11.	9.	11.	31.
CONFIGURATION CONTROL	3.	2.	14.	18.
DOCUMENTATION	2.	0.	4.	6.
PROGRAM MANAGEMENT	4.	1.	2.	7.
TOTAL	50.	14.	58.	122.

ADDITIONAL DATA

DESCRIPTORS

INSTRUCTIONS	70130	APPLICATION	1.588	RESOURCE	1.683
FUNCTIONS	229	STRUCTURE	0.000	LEVEL	0.000

SCHEDULE

COMPLEXITY	0.800				
DESIGN START	NOV 75	IMPL START	0	T&I START	0
DESIGN END	0	IMPL END	0	T&I END	AUG 78

--- PRICE SOFTWARE MODEL ---

ECIRP SPECIFICATIONS MODE

DATE 16-OCT-79 TIME 19:29

PROJECT BOB WARTIME SIMULATION MODEL

PROGRAM COSTS

COST ELEMENTS	DESIGN	IMPL	T & I	TOTAL
SYSTEMS ENGINEERING	10.	1.	8.	19.
PROGRAMMING	2.	3.	3.	9.
CONFIGURATION CONTROL	1.	1.	4.	6.
DOCUMENTATION	1.	0.	1.	2.
PROGRAM MANAGEMENT	1.	0.	1.	2.
TOTAL	15.	5.	17.	37.

ADDITIONAL DATA

DESCRIPTORS

INSTRUCTIONS	30000	APPLICATION	1.588	RESOURCE	1.454
FUNCTIONS	333	STRUCTURE	0.000	LEVEL	0.000

SCHEDULE

COMPLEXITY	1.000				
DESIGN START	OCT 77	IMPL START	0	T&I START	0
DESIGN END	0	IMPL END	0	T&I END	AUG 78

--- PRICE SOFTWARE MODEL ---

ECIRP SPECIFICATIONS MODE

DATE 16-OCT-79 TIME 19:29

PROJECT BOB MAINTENANCE SIMULATION SYSTEM

PROGRAM COSTS

COST ELEMENTS	DESIGN	IMPL	T & I	TOTAL
SYSTEMS ENGINEERING	2.	0.	2.	4.
PROGRAMMING	1.	1.	1.	2.
CONFIGURATION CONTROL	0.	0.	1.	1.
DOCUMENTATION	0.	0.	0.	0.
PROGRAM MANAGEMENT	0.	0.	0.	0.
TOTAL	3.	1.	4.	8.

ADDITIONAL DATA

DESCRIPTORS

INSTRUCTIONS	2300	APPLICATION	1.588	RESOURCE	1.738
FUNCTIONS	32	STRUCTURE	0.000	LEVEL	0.000

SCHEDULE

COMPLEXITY	1.000				
DESIGN START	FEB 78	IMPL START	0	T&I START	0
DESIGN END	0	IMPL END	0	T&I END	JUL 78

--- PRICE SOFTWARE MODEL ---

DATE 18-OCT-79 TIME 21:10

PROJECT DCB — OPERATIONAL PLANNING SYSTEM

PROGRAM COSTS

COST ELEMENTS	DESIGN	IMPL	T & I	TOTAL
SYSTEMS ENGINEERING	99.	6.	92.	197.
PROGRAMMING	21.	32.	38.	91.
CONFIGURATION CONTROL	15.	8.	54.	77.
DOCUMENTATION	10.	2.	16.	28.
PROGRAM MANAGEMENT	11.	2.	10.	23.
TOTAL	156.	50.	210.	416.

ADDITIONAL DATA

DESCRIPTORS

INSTRUCTIONS	315000	APPLICATION	1.588	RESOURCE	1.500
FUNCTIONS	3500	STRUCTURE	0.000	LEVEL	0.000

SCHEDULE

COMPLEXITY	1.000				
DESIGN START	FEB 78	IMPL START	0	T&I START	0
DESIGN END	0	IMPL END	0	T&I END	MAR 82

SENSITIVITY DATA

COMPLEXITY

RESOURCE		.900	1.000	1.100
1.400	COST	376.	375.	376.
	MONTHS	49.0	49.0	49.0
1.500	COST	418.	416.	417.
	MONTHS	49.0	49.0	49.0
1.600	COST	461.	459.	460.
	MONTHS	49.0	49.0	49.0

--- PRICE SOFTWARE MODEL ---

DATE 16-OCT-79 TIME 21:12

PROJECT OCB OPERATIONAL PLANNING SYSTEM

SCHEDULE EFFECT SUMMARY

ACTIVITY LENGTH IN MONTHS

COMPLEXITY = 1.000	DESIGN	IMPL	T & I	TOTAL
SPECIFIED SCHEDULE	0.0	0.0	0.0	49.0
(OVERLAP)	(0.0)	(0.0)		
TYPICAL SCHEDULE	4.8	3.1	4.8	11.4
(OVERLAP)	(0.2)	(0.5)		

DEVELOPMENT COSTS

COMPLEXITY = 1.000	DESIGN	IMPL	T & I	TOTAL
SPECIFIED SCHEDULE	156.	50.	210.	416.
TYPICAL SCHEDULE	92.	29.	115.	236.
ESTIMATED PENALTY	64.	21.	96.	181.

--- PRICE SOFTWARE MODEL ---

DATE 18-OCT-79 TIME 20:59

PROJECT DCB ENGINEERING MANAGEMENT SYSTEM

PROGRAM COSTS

COST ELEMENTS	DESIGN	IMPL	T & I	TOTAL
SYSTEMS ENGINEERING	20.	1.	17.	38.
PROGRAMMING	4.	6.	7.	17.
CONFIGURATION CONTROL	2.	1.	9.	12.
DOCUMENTATION	2.	0.	2.	4.
PROGRAM MANAGEMENT	2.	0.	2.	4.
TOTAL	30.	9.	37.	76.

ADDITIONAL DATA

DESCRIPTORS

INSTRUCTIONS	66000	APPLICATION	1.588	RESOURCE	1.500
FUNCTIONS	233	STRUCTURE	0.000	LEVEL	0.000

SCHEDULE

COMPLEXITY	1.000				
DESIGN START	JAN 75	IMPL START	0	T&I START	0
DESIGN END	0	IMPL END	0	T&I END	DEC 76

SENSITIVITY DATA

COMPLEXITY

RESOURCE		.900	1.000	1.100
1.400	COST	69.	69.	69.
	MONTHS	23.0	23.0	23.0
1.500	COST	76.	76.	76.
	MONTHS	23.0	23.0	23.0
1.600	COST	83.	82.	82.
	MONTHS	23.0	23.0	23.0

--- PRICE SOFTWARE MODEL ---

DATE 16-OCT-79 TIME 21:01

PROJECT DCB ENGINEERING MANAGEMENT SYSTEM

SCHEDULE EFFECT SUMMARY

ACTIVITY LENGTH IN MONTHS

COMPLEXITY = 1.000	DESIGN	IMPL	T & I	TOTAL
SPECIFIED SCHEDULE	0.0	0.0	0.0	23.0
%OVERLAP		(0.0)	(0.0)	
TYPICAL SCHEDULE	2.7	1.6	2.6	6.3
%OVERLAP		(0.3)	(0.2)	

DEVELOPMENT COSTS

COMPLEXITY = 1.000	DESIGN	IMPL	T & I	TOTAL
SPECIFIED SCHEDULE	30.	9.	37.	76.
TYPICAL SCHEDULE	20.	6.	24.	51.
ESTIMATED PENALTY	9.	3.	13.	25.

--- PRICE SOFTWARE MODEL ---

DATE 16-OCT-79 TIME 21:15

PROJECT ACB MODIFICATION MANAGEMENT SYSTEM

PROGRAM COSTS

COST ELEMENTS	DESIGN	IMPL	T & T	TOTAL
SYSTEMS ENGINEERING	10.	1.	8.	18.
PROGRAMMING	2.	3.	3.	8.
CONFIGURATION CONTROL	1.	1.	4.	5.
DOCUMENTATION	1.	0.	1.	2.
PROGRAM MANAGEMENT	1.	0.	1.	2.
TOTAL	14.	4.	17.	36.

ADDITIONAL DATA

DESCRIPTORS

INSTRUCTIONS	27900	APPLICATION	1.588	RESOURCE	1.500
FUNCTIO	310	STRUCTURE	0.000	LEVEL	0.000

SCHEDULE

COMPLEXITY	1.000				
DESIGN START	MAY 78	IMPL START	0	T&T START	0
DESIGN END	0	IMPL END	0	T&T END	JAN 79

SENSITIVITY DATA

COMPLEXITY

RESOURCE		.900		1.000		1.100	
1.400	COST	33.	COST	33.	COST	33.	
	MONTHS	8.0	MONTHS	8.0	MONTHS	8.0	
1.500	COST	35.	COST	36.	COST	36.	
	MONTHS	8.0	MONTHS	8.0	MONTHS	8.0	
1.600	COST	38.	COST	39.	COST	39.	
	MONTHS	8.0	MONTHS	8.0	MONTHS	8.0	

--- PRICE SOFTWARE MODEL ---

DATE 18-OCT-79 TIME 21:17

PROJECT ACB MODIFICATION MANAGEMENT SYSTEM

SCHEDULE EFFECT SUMMARY

ACTIVITY LENGTH IN MONTHS

COMPLEXITY = 1.000	DESIGN		IMPL		T & I	TOTAL
SPECIFIED SCHEDULE	0.0		0.0		0.0	8.0
%OVERLAP		% 0.0%		% 0.0%		
TYPICAL SCHEDULE	1.9		1.1		1.8	4.4
%OVERLAP		% 0.2%		% 0.1%		

DEVELOPMENT COSTS

COMPLEXITY = 1.000	DESIGN	IMPL	T & I	TOTAL
SPECIFIED SCHEDULE	14.	4.	17.	36.
TYPICAL SCHEDULE	13.	4.	15.	33.
ESTIMATED PENALTY	1.	0.	1.	3.

--- PRICE SOFTWARE MODEL ---

DATE 18-OCT-79 TIME 21:20

PROJECT ACB MATERIAL REPORTING SYSTEM

PROGRAM COSTS

COST ELEMENTS	DESIGN	IMPL	T & I	TOTAL
SYSTEMS ENGINEERING	26.	1.	23.	51.
PROGRAMMING	6.	8.	10.	23.
CONFIGURATION CONTROL	3.	2.	12.	17.
DOCUMENTATION	2.	0.	3.	6.
PROGRAM MANAGEMENT	2.	1.	2.	5.
TOTAL	40.	12.	50.	102.

ADDITIONAL DATA

DESCRIPTORS

INSTRUCTIONS	20130	APPLICATION	1.588	RESOURCE	1.500
FUNCTIONS	229	STRUCTURE	0.000	LEVEL	0.000

SCHEDULE

COMPLEXITY	1.000				
DESIGN START	NOV 75	IMPL START	0	T&I START	0
DESIGN END	0	IMPL END	0	T&I END	AUG 78

SENSITIVITY DATA

COMPLEXITY

RESOURCE		.900	1.000	1.100
1.400	COST	93.	93.	93.
	MONTHS	33.0	33.0	33.0
1.500	COST	103.	102.	102.
	MONTHS	33.0	33.0	33.0
1.600	COST	112.	111.	111.
	MONTHS	33.0	33.0	33.0

--- PRICE SOFTWARE MODEL ---

DATE 18-OCT-79 TIME 21:21

PROJECT ACB MATERIAL REPORTING SYSTEM

SCHEDULE EFFECT SUMMARY

ACTIVITY LENGTH IN MONTHS

COMPLEXITY = 1.000	DESIGN	IMPL	T & I	TOTAL
SPECIFIED SCHEDULE	0.0	0.0	0.0	33.0
(OVERLAP)	% 0.0%	% 0.0%		
TYPICAL SCHEDULE	2.7	1.6	2.6	6.4
(OVERLAP)	% 0.3%	% 0.2%		

DEVELOPMENT COSTS

COMPLEXITY = 1.000	DESIGN	IMPL	T & I	TOTAL
SPECIFIED SCHEDULE	40.	12.	50.	102.
TYPICAL SCHEDULE	23.	7.	27.	52.
ESTIMATED PENALTY	17.	6.	23.	45.

--- PRICE SOFTWARE MODEL ---

DATE 16-OCT-79 TIME 21:27

PROJECT AOB — WARTIME SIMULATION MODEL

PROGRAM COSTS

COST ELEMENTS	DESIGN	IMPL	T & T	TOTAL
SYSTEMS ENGINEERING	10.	1.	9.	20.
PROGRAMMING	2.	3.	4.	9.
CONFIGURATION CONTROL	1.	1.	4.	6.
DOCUMENTATION	1.	0.	1.	2.
PROGRAM MANAGEMENT	1.	0.	1.	2.
TOTAL	16.	5.	18.	38.

ADDITIONAL DATA

DESCRIPTORS

INSTRUCTIONS	30000	APPLICATION	1.588	RESOURCE	1.500
FUNCTIONS	333	STRUCTURE	0.000	LEVEL	0.000

SCHEDULE

COMPLEXITY	1.000				
DESIGN START	OCT 77	IMPL START	0	T&T START	0
DESIGN END	0	IMPL END	0	T&T END	AUG 78

SENSITIVITY DATA

COMPLEXITY

RESOURCE		.900	1.000	1.100
1.400	COST	35.	35.	35.
	MONTHS	10.0	10.0	10.0
1.500	COST	39.	38.	39.
	MONTHS	10.0	10.0	10.0
1.600	COST	42.	42.	42.
	MONTHS	10.0	10.0	10.0

--- PRICE SOFTWARE MODEL ---

DATE 16-OCT-79 TIME 21:26

PROJECT 808 WARTIME SIMULATION MODEL

SCHEDULE EFFECT SUMMARY

ACTIVITY LENGTH IN MONTHS

COMPLEXITY = 1.000	DESIGN	IMPL	T & I	TOTAL
SPECIFIED SCHEDULE	0.0	0.0	0.0	10.0
%OVERLAP	% 0.0%	% 0.0%		
TYPICAL SCHEDULE	1.9	1.1	1.8	4.6
%OVERLAP	% 0.2%	% 0.1%		

DEVELOPMENT COSTS

COMPLEXITY = 1.000	DESIGN	IMPL	T & I	TOTAL
SPECIFIED SCHEDULE	16.	5.	18.	38.
TYPICAL SCHEDULE	13.	4.	16.	33.
ESTIMATED PENALTY	2.	1.	3.	5.

--- PRICE SOFTWARE MODEL ---

DATE 16-OCT-79 TIME 21:28

PROJECT MOB — MAINTENANCE SIMULATION SYSTEM

PROGRAM COSTS

COST ELEMENTS	DESIGN	IMPL	T & I	TOTAL
SYSTEMS ENGINEERING	2.	0.	2.	4.
PROGRAMMING	0.	1.	1.	2.
CONFIGURATION CONTROL	0.	0.	1.	1.
DOCUMENTATION	0.	0.	0.	0.
PROGRAM MANAGEMENT	0.	0.	0.	0.
TOTAL	3.	1.	3.	7.

ADDITIONAL DATA

DESCRIPTORS

INSTRUCTIONS	2900	APPLICATION	1.588	RESOURCE	1.500
FUNCTIONS	32	STRUCTURE	0.000	LEVEL	0.000

SCHEDULE

COMPLEXITY	1.000				
DESIGN START	FEB 78	IMPL START	0	T&I START	0
DESIGN END	0	IMPL END	0	T&I END	JUL 78

SENSITIVITY DATA

COMPLEXITY

RESOURCE		.900		1.000		1.100	
1.400	COST	6.	COST	6.	COST	6.
	MONTHS	5.0	MONTHS	5.0	MONTHS	5.0
1.500	COST	7.	COST	7.	COST	7.
	MONTHS	5.0	MONTHS	5.0	MONTHS	5.0
1.600	COST	7.	COST	7.	COST	7.
	MONTHS	5.0	MONTHS	5.0	MONTHS	5.0

--- PRICE SOFTWARE MODEL ---

DATE 16-OCT-79 TIME 21:30

PROJECT AOB MAINTENANCE SIMULATION SYSTEM

SCHEDULE EFFECT SUMMARY

ACTIVITY LENGTH IN MONTHS

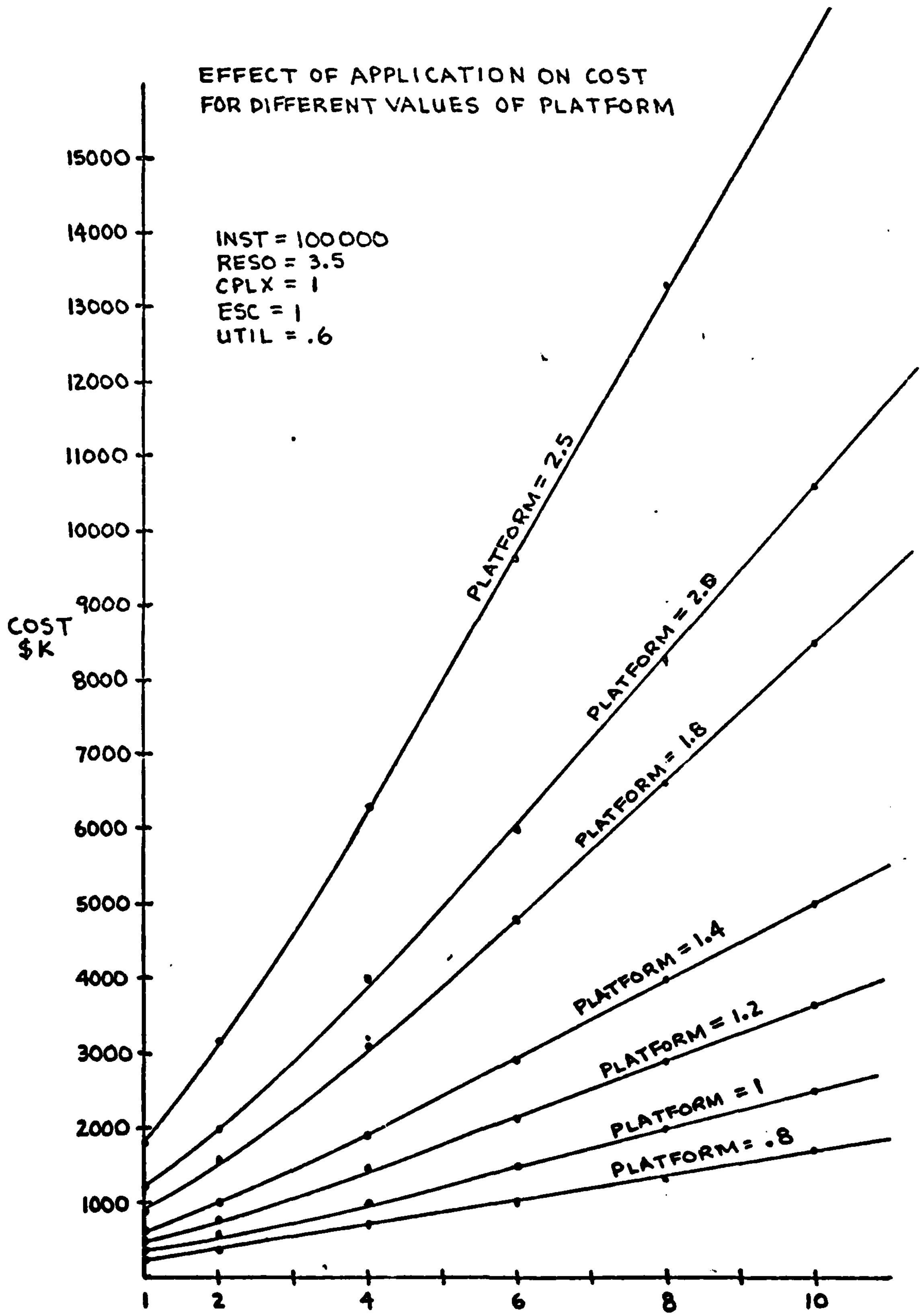
COMPLEXITY = 1.000	DESIGN		IMPL		T & I	TOTAL
SPECIFIED SCHEDULE	0.0		0.0		0.0	5.0
%OVERLAP		% 0.0%		% 0.0%		
TYPICAL SCHEDULE	0.8		0.5		0.8	1.9
%OVERLAP		% 0.1%		% 0.1%		

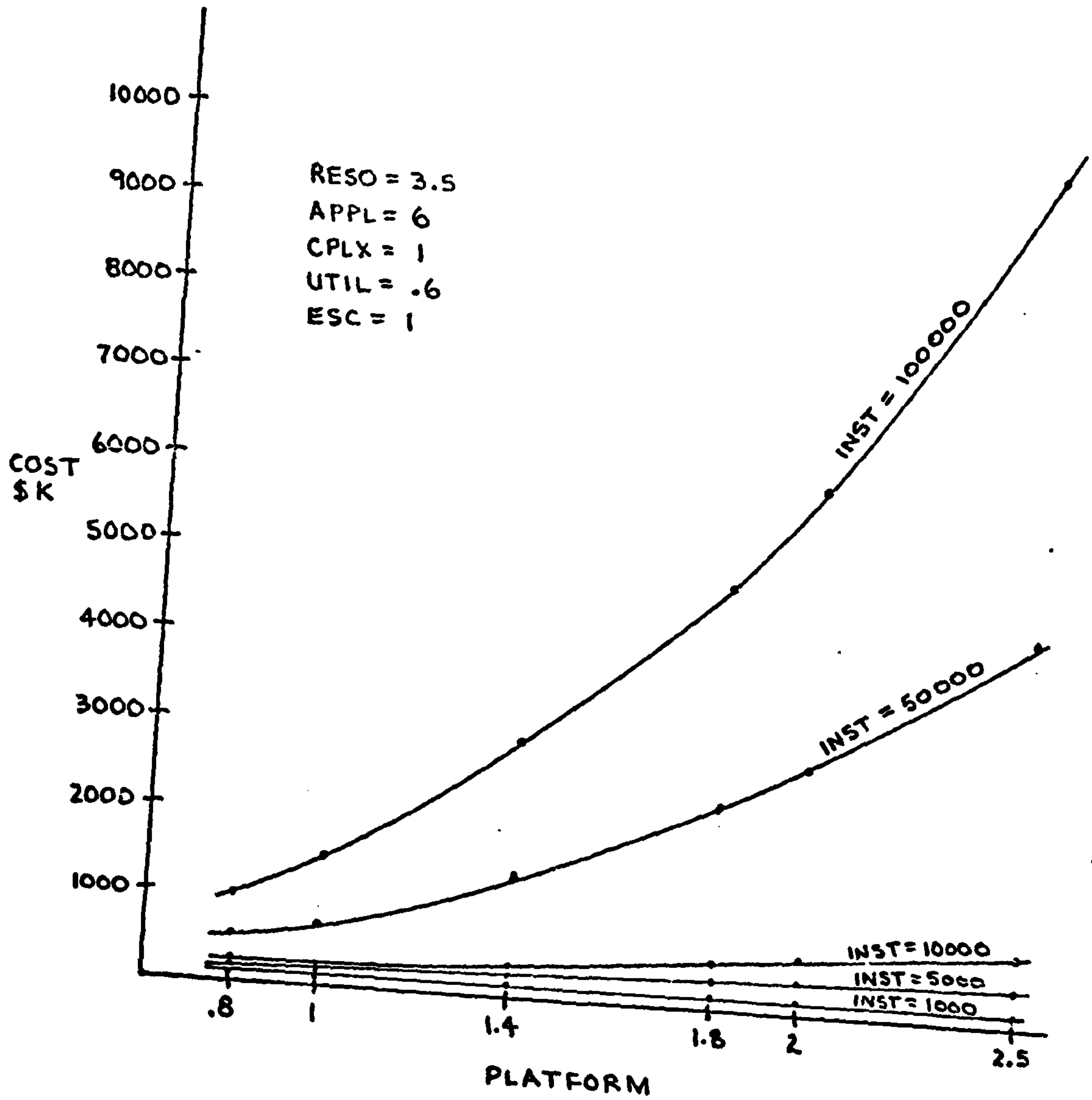
DEVELOPMENT COSTS

COMPLEXITY = 1.000	DESIGN	IMPL	T & I	TOTAL
SPECIFIED SCHEDULE	3.	1.	3.	7.
TYPICAL SCHEDULE	2.	1.	3.	6.
ESTIMATED PENALTY	0.	0.	0.	1.

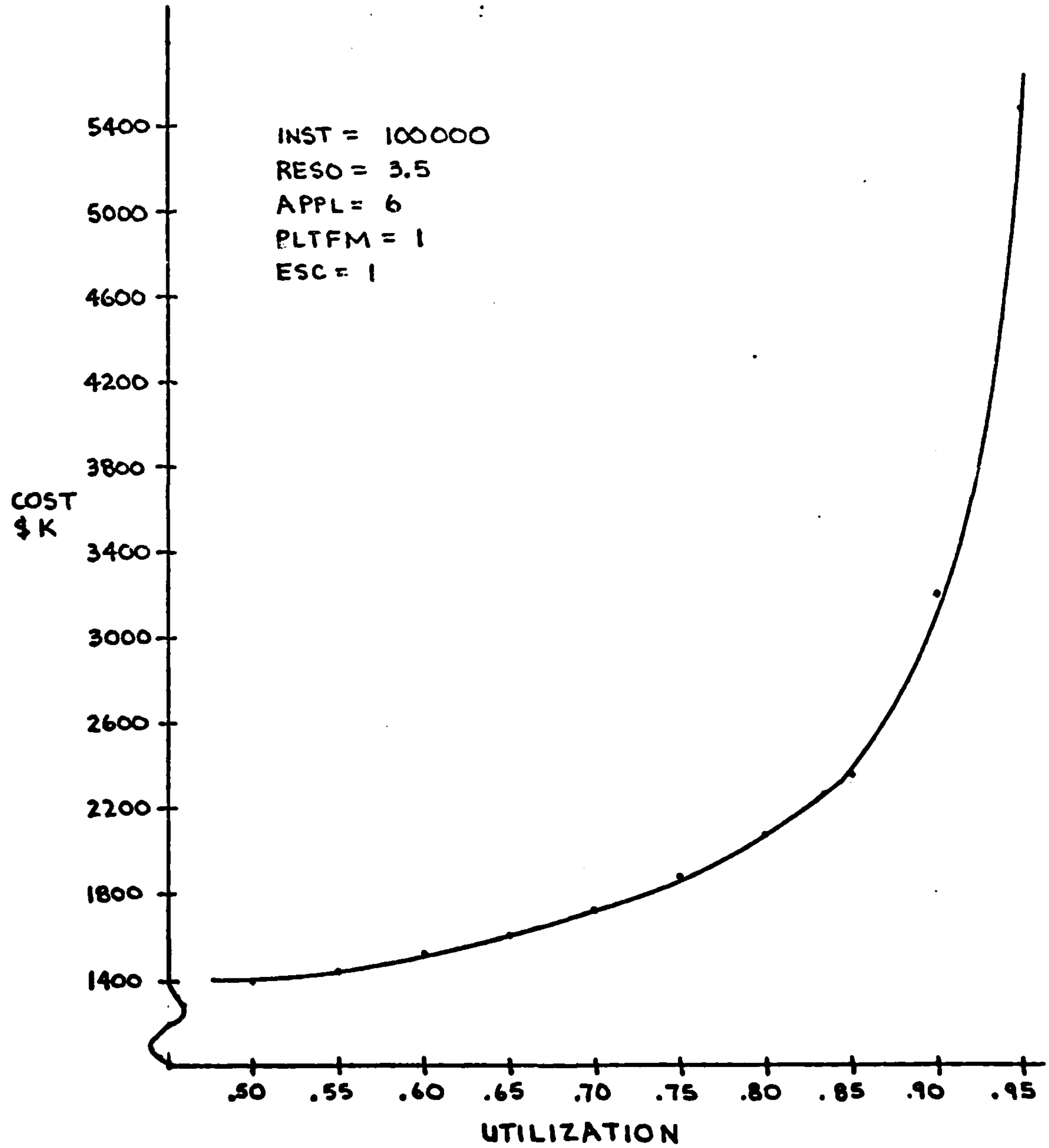
APPENDIX C

Sensitivity Analysis Results

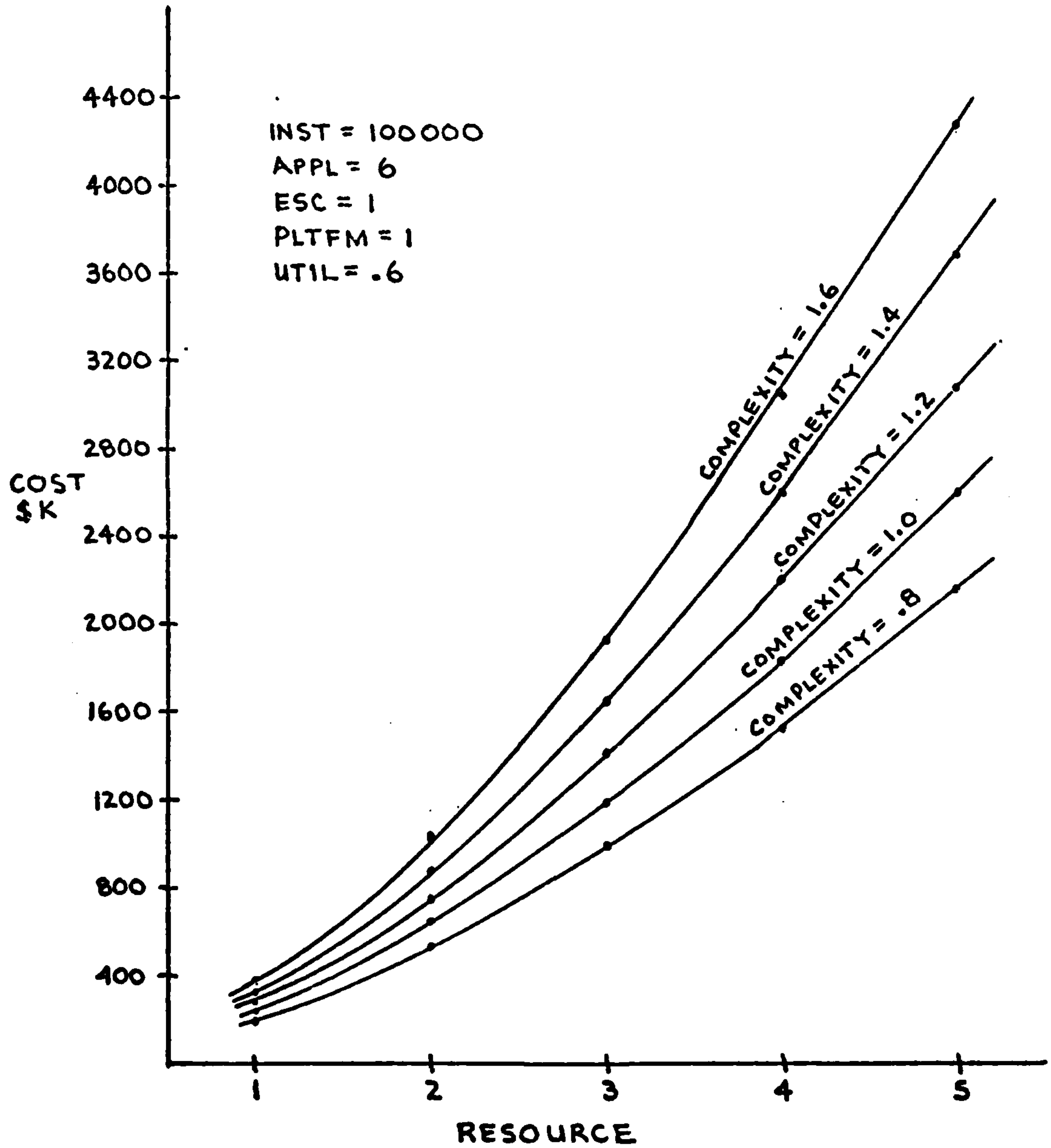

EFFECT OF APPLICATION ON COST
FOR DIFFERENT VALUES OF PLATFORM
INST = 100000
RESO = 3.5
CPLX = 1
ESC = 1
UTIL = .6
COST $K
15000
14000
13000
12000
11000
10000
9000
8000
7000
6000
5000
4000
3000
2000
1000
PLATFORM = 2.5
PLATFORM = 2.0
PLATFORM = 1.8
PLATFORM = 1.4
PLATFORM = 1.2
PLATFORM = 1
PLATFORM = .8
1
2
4
6
8
10
APPLICATION



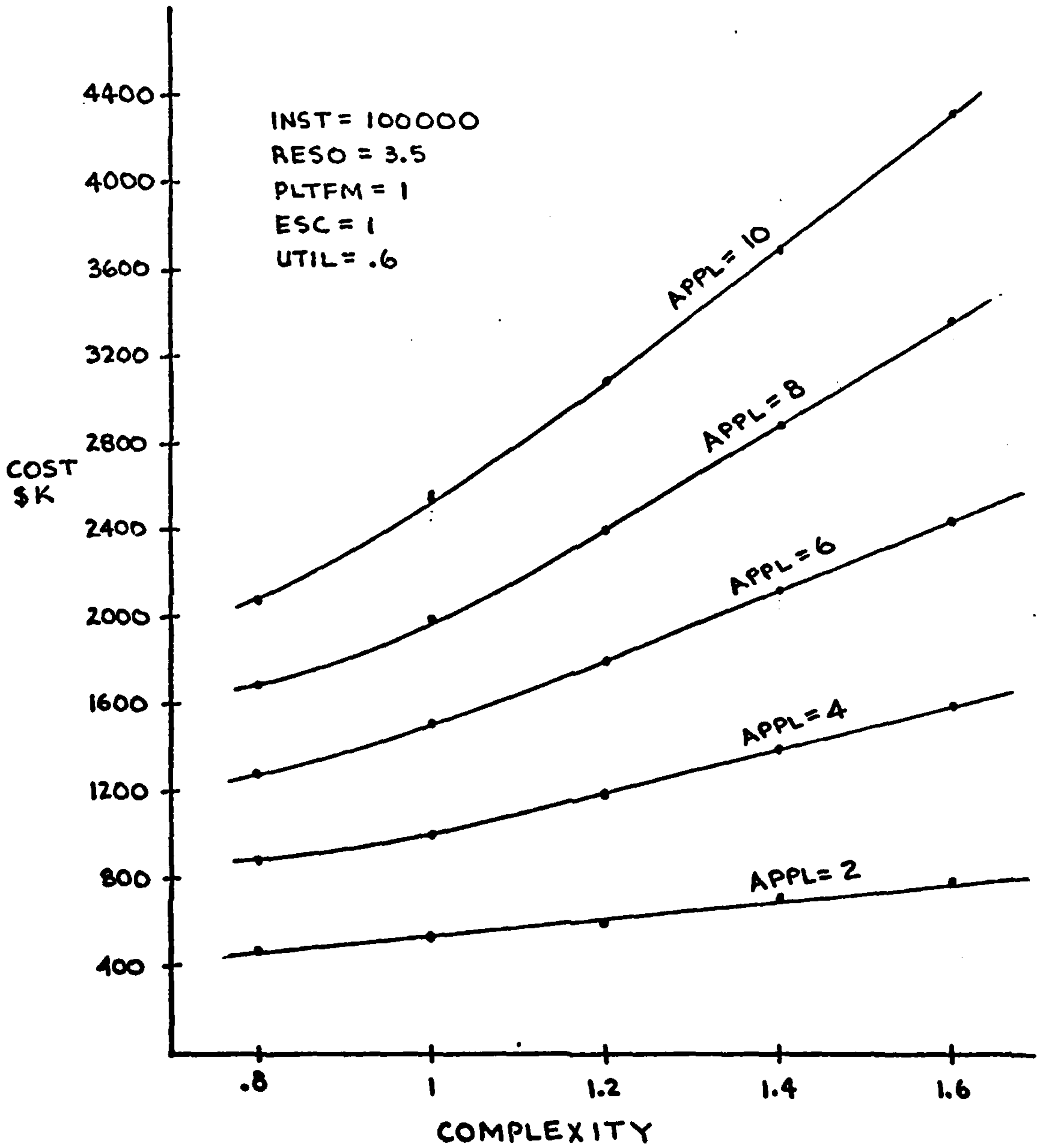
EFFECT OF PLATFORM ON COST
FOR DIFFERENT VALUES OF INSTRUCTION



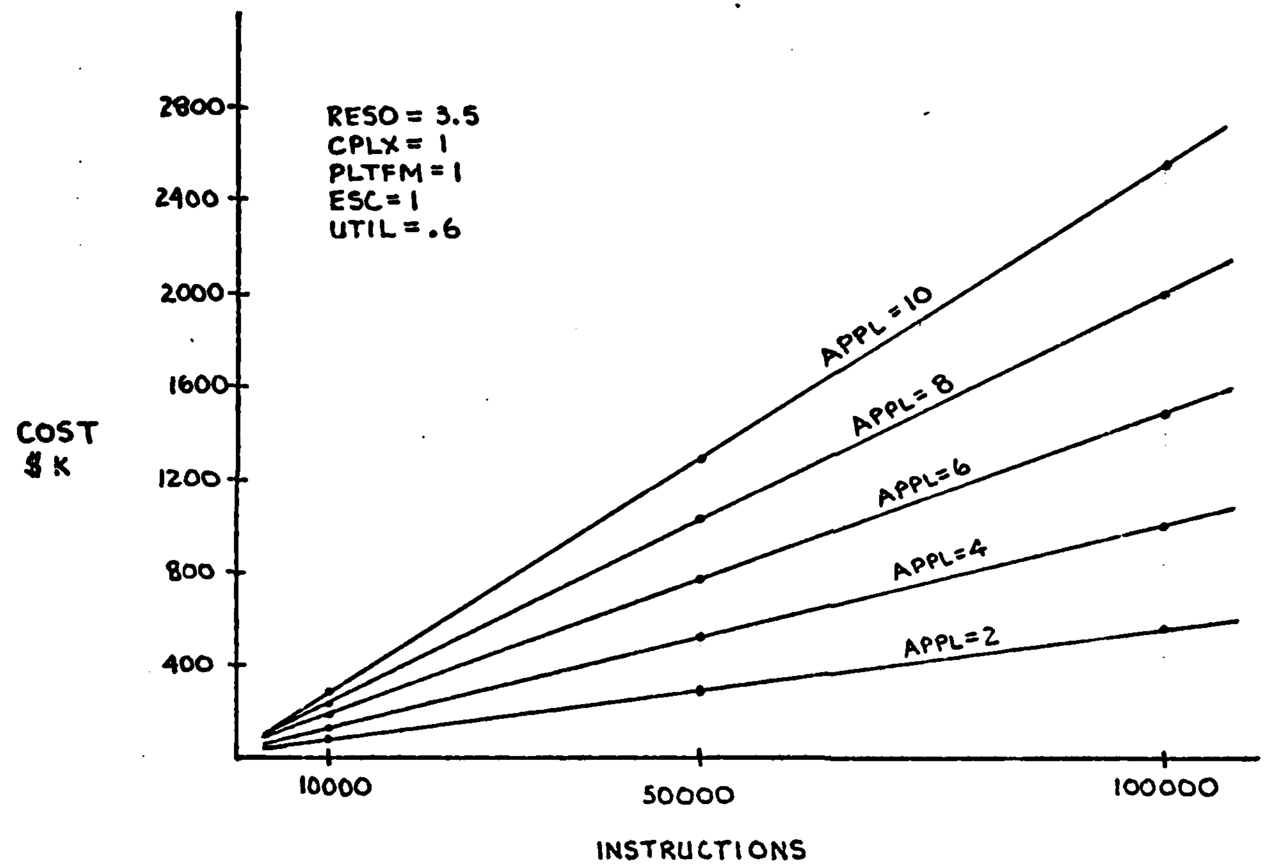
EFFECT OF UTILIZATION ON COST



EFFECT OF RESOURCE ON COST
FOR DIFFERENT VALUES OF COMPLEXITY



EFFECT OF COMPLEXITY ON COST
FOR DIFFERENT VALUES OF APPLICATION



EFFECT OF INSTRUCTIONS ON COST
FOR DIFFERENT VALUES OF APPLICATION

Vita

Captain Raymond Steffey Jr. was born on August 22, 1944 in Baltimore, Maryland. He attended Johns Hopkins University prior to enlisting in the U.S. Air Force in 1965. From 1965 through 1967 he was a technical instructor in the 465L computer maintenance course at Keesler AFB, Mississippi. In 1967 Captain Steffey was selected to attend Oklahoma State University under the Airman Education and Commissioning Program. He received a Bachelor of Science degree in Industrial Engineering and Management in 1970, and subsequently attended the Air Force Officer Training School being commissioned as a second lieutenant upon graduation.

From 1970 through 1973, Captain Steffey was assigned to the Directorate of Data Automation Headquarters Aerospace Defense Command in Colorado Springs, Colorado where he was responsible for the command-wide implementation of the Base Level B-3500 computer system. During the period 1974 to 1978 he was assigned to the Office of the Assistant for Automation, DCS Plans and Operations, Headquarters United States Air Force at the Pentagon, where he was responsible for monitoring operational applications of Air Force computer system development programs.

Captain Steffey is currently assigned to the School of Engineering, Air Force Institute of Technology, where he is

pursuing a Master of Science degree in Systems Management.

Captain Steffey is married to the former Miss Sharon Gautney of Baltimore, Msryland. They have two children, Laura and Nicole.

Permanent Address: 5925 Lillyan Ave.
Baltimore, MD 21206

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER AFIT/GSM/SM/79D-20	2. GOVT ACCESSION NO. AD-A083 713	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) AN ANALYSIS OF THE RCA PRICE-S COST ESTIMATION MODEL AS IT RELATES TO CURRENT AIR FORCE COMPUTER SOFTWARE ACQUISITION AND MANAGEMENT		5. TYPE OF REPORT & PERIOD COVERED Master's Thesis
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Raymond E. Steffey Jr. Captain USAF		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Air Force Institute of Technology (AFIT/EN) Wright-Patterson AFB OH 45433		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS		12. REPORT DATE December 1979
		13. NUMBER OF PAGES 232
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)		15. SECURITY CLASS. (of this report) UNCLASSIFIED
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

Approved for public release; IAW AFR 190-17

JOSEPH P. HIPPS, Major, USAF
Director of Information

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

PRICE-S System
Software
Cost Estimation
Computer Resource Management
Embedded Software Systems
General Purpose Software

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

The enormous technical accomplishments of the computer industry have led to the building of computers of all sizes and complexities. As the range of defense computer applications grows and the complexity of the tasks these systems are called upon to handle increases, the costs of developing the application software has also increased such that it has now become a dominant component in the total system cost.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

Block 20:

Many software acquisitions have experienced cost and schedule overruns leading to unanticipated cost growth. These experiences have highlighted the need to improve methods of software cost estimation. Software cost estimation is essential to budgeting, allocation of resources, and control of expenditures throughout the life cycle of a system. Accurate predictions of software costs are required in order to make practical and realistic tradeoffs between system capabilities and life cycle costs. The purpose of this research is to provide those involved in the software cost estimation task with an introduction to Air Force computer resource acquisition and management in general; and specifically to investigate the applicability of the RCA PRICE-S software cost estimation model to Air Force applications system development. A mass of computer software acquisition and management study, policy, and guidance literature was reviewed, and an attempt was made to consolidate the most pertinent information into a description of the overall processes. Historical cost and schedule data were collected on 18 Air Force software development projects. This data which included systems of the three major Air Force applications areas of: 1) embedded avionics; 2) embedded command and control; and 3) management data systems was used to calibrate and validate the PRICE-S cost estimation model.

Based on the data available in this preliminary analysis effort, it appears that the PRICE-S model is compatible with current Air Force software acquisition and management techniques. A system such as the PRICE-S system, combined with an adequate data collection methodology, might be successfully implemented giving the Air Force the capability to accurately predict and track future software development costs across the entire spectrum of software applications.